Extending Controlnet Pose Control to Quadrupeds

David Booth
Princeton University
dbooth@princeton.edu

Abstract

This paper presents a new pre-trained model for the ControlNet architecture which uses quadruped pose as a parameter to guide the generation of Stable Diffusion images. This model was trained using a small dataset (< 8k) of quadruped poses of 54 different species using images and pose keypoints from the AP-10k dataset. A new quadruped skeleton is used for this model, replacing the previously used biped skeleton referenced in ControlNet's implementation of Openpose. Even though this model's dataset size of < 10k images is relatively small, this model is still able to successfully use quadruped pose as a control input with robust results, creating a pre-trained model which successfully uses animal pose estimates to control outputs of Stable Diffusion at a relatively low quality. Future work using this paper's method will be needed to create better datasets and train this model for more iterations before a production-quality model is ready. This project can be found at: https://github.com/abehonest/ControlNet_AnimalPose/

1. Introduction

The ControlNet architecture allows users to guide image generation within diffusion models using a control image. In ControlNet's original paper, the authors proposed 8 distinct implementations of its architecture. One such implementation, based on Openpose's human pose estimation model, takes an image of the pose-estimated skeleton of a human and uses it to guide Stable Diffusion's image generation to create a final image with a humanoid figure in a pose similar to the pose shown in the control image.[13] This implementation only works for humanoid figures and does not satisfactorily work for quadrupeds. Using a normal bipedal pose as a control input and prompting Stable Diffusion to generate a quadruped in that pose does not generally generate quadrupeds in bipedal poses, instead generating a humanoid in the pose and the prompted quadruped nearby. Using the bipedal skeleton in a quadruped-like position, accomplished by inputting a picture of a human on all-fours, generates quadrupeds in roughly the same position as the skeleton, however their limbs and overall pose generally do not match the control input. Examples of inferences run using these approaches are shown in Figures 1 and 2. This paper describes a new implementation for ControlNet, trained to use a new skeleton, based on the AP-10k dataset's skeleton model, to use quadruped poses to guide Stable Diffusion image generation.

2. Related Work

2.1. Stable Diffusion

Stable Diffusion is an open source text-to-image diffusion model capable of creating high-quality output images using a natural language input. This paper trains ControlNet with Stable Diffusion as a base model.[9]

2.2. ControlNet

ControlNet is an architecture which can be used to guide the specific placement of certain aspects of a diffusion model's output image. It is a neural network which can be paired with a pre-trained image diffusion model with locked weights. ControlNet can be used to influence aspects of the diffusion network's generated image without retraining the diffusion network itself. ControlNet uses several "zero-convolution" layers, which are just normal convolutional layers with their weights always initialized to zero, which guarantee that ControlNet will start out training with minimal effect on a diffusion network's generated images and slowly learn the weights for its zero-convolution layers until the desired control effect is reached. Because ControlNet only builds upon existing diffusion models without changing their weights a given ControlNet can be trained with a relatively small dataset while still retaining much of the original level of quality from the original diffusion model. ControlNet has several pre-made implementations listed on its paper, two of which involve bipedal pose estimation. The Openpifpaf and Openpose implementations use differing pose models to generate images using human pose as a control parameter. [13]

2.3. ControlLoRA

Control LoRA uses the LoRA (Low Rank Adaption) model to lower the model size of both the ControlNet itself and its associated diffusion model. LoRA takes a frozen NLP model and "injects trainable rank decomposition matrices into each layer of the Transformer architecture, greatly reducing the number of trainable parameters for downstream tasks." This allows for the generation of a small ControlNet with minimal loss in quality which allows ease in sending pre-trained models over the internet due to their low overall size.[4][10]

2.4. Openpose

Openpose is a human pose estimation model trained on the COCO dataset. It is able to estimate the poses of multiple humans in a given frame, finding keypoints for 18 distinct parts of the human body. Openpose has three different models: a body and foot model, a hand model, and a face model. In conjunction, these three models can generate a pose estimate for an entire human body, the position of its hands and fingers, and the expression on its face. ControlNet was trained on a dataset of 200,000 images of humans and their corresponding pose estimations generated by Openpose to create a model which uses Openpose human pose estimates to control the position of human subjects in generated images. ControlNet was trained on a complete pose estimate using the body, hand, and face models, however inputs using only one of said models are sufficient to generate a pose constrained with only the parameters of that single model. For example, a pose can be generated using only the output of the body model without the hand and face model, allowing Stable Diffusion to 'fill in the gaps'.[3]

2.5. Openpifpaf

Openpifpaf is another human pose estimation model which ControlNet was trained on. It uses a different skeleton from Openpose with 17 keypoints. Its implementation for ControlNet was trained on 80,000 images of humans and their corresponding Openpifpaf pose estimations with one major filtering rule: Poses with 30 percent or less keypoints were filtered out and not used in training.[5]

2.6. AP-10k

The Animal Pose 10k (AP-10k) dataset is a collection of 10,015 animal pictures scraped from the internet of 54 different species and their corresponding ground truth pose keypoints. These ground truth pose keypoints were manually selected and checked by a group of 13 trained annotators. Each image within AP-10k has at least one animal in it, with multiple instances of animals getting labeled with multiple sets of keypoints such that each animal in a given image has an associated pose skeleton. 17 keypoints are identified for each animal to create a unique skeleton which works well for varied quadruped species and differs from the human skeleton used in Openpose. AP-10k's

skeleton uses a keypoint for the tailbone of a given animal and separate keypoints for the animal's hips, likely due to the variance in quadruped tailbone/hip size. Openpose also uses two keypoints per eye, one for each side of the eye, while AP-10k uses a single keypoint per eye.[12]

The AP-10k dataset was used to train a pose estimation model within the MMPose pose estimation library. This model is publicly available and can be used to infer AP-10k keypoints on any animal picture.[8]

2.7. Animals with Attributes 2

Animals with Attributes 2 (AwA2) is a dataset of roughly 37,000 animal images with both animal species and attributes of a given animal labeled. It contains no pose labels.[11]

2.8. Animal Pose

Animal Pose is an animal pose estimation dataset with roughly 6,000 annotated images of 5 different species. It was published prior to AP-10k, which wished to build upon it and create a larger animal dataset with a larger number of distinct species labeled. Animal pose uses a skeleton similar to AP-10k's skeleton, differing in three keypoints. Animal Pose adds two keypoints to define the position of each animal's ear and one keypoint to define the position of each animal's scruff (back of neck).[2]

2.9. Horse-10

Horse-10 is a dataset which contains 8,114 images of horses with corresponding pose annotations. Horse-10's dataset contains images of 30 different breeds of horses in different poses. Most of the poses within Horse-10 are frames within a video taken of a given horse walking. Because of this Horse-10 has a considerably lower variance in subject and background between pictures than other animal pose datasets mentioned in this paper.[7]

2.10. BLIP

Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation (BLIP) is a publicly accessible language model which can take an image as an input and describe characteristics of it. BLIP can be used to generate succinct and accurate captions for a wide variety of images.[6]

3. Method

In order to train a ControlNet model, four inputs are required. First, a pre-trained diffusion model must be loaded for ControlNet to inference on. For this implementation Stable Diffusion 1.5 is used. Next, the three training inputs are required: a 'control image', a ground truth image, and a prompt describing the ground truth image. The control image must have some form of pre-determined reproducible structure so that ControlNet can learn in what way the control image corresponds to its ground truth image. This paper uses the AP-10k dataset and MMPose model for generating training and validation control skeleton images and providing ground truth images.

To generate a control input and ground truth pair, the AP-10k dataset's animal photos are used as ground truth images and, using AP-10k's provided keypoint annotations to guide keypoint placement, a corresponding control skeleton is generated on a black background. Both images are resized and cropped to a resolution of 512x512 as required by ControlNet for input images. The control skeleton uses AP-10k's skeleton model, drawing lines between each connected keypoint using OpenCV. These lines were each given a color value such that each line's color is roughly 100 units on an 8-bit RGB scale different (using euclidean distance) from the next closest line, with exception to the back's line and front-left calve's line which are similar due to a mis-input in values which was not noticed until after training. This mistake does not seem to have any major affect on the model, and actually mirrors Openpose's skeletal images which have some color overlap between limbs. Images with one or more animals were allowed into the training dataset. The only filtering rule used was Controlnet's Openpifpaf Implementation's filtering rule of only allowing a given skeleton to be generated if more that 30 percent of its keypoints were

available.[13] This lowered the dataset size from 10,015 images to 7,465 images, however this was still sufficient to reach convergence on a model.

After the generation of all control/ground truth pairs, prompts are generated using the same method as the ControlNet paper. ControlNet, and this paper, use the BLIP model to automatically generate succinct and descriptive captions for ground truth images.

The model was trained for 20 hours on an Nvidia A10 GPU, allowing for 22 epochs of training and a total of 89,582 iterations at batch size 2. The model suddenly converged around Epoch 8.

4. Analysis

Several different training datasets were tried with varying results.

4.1. Failed Methods

Initially this paper attempted to use the Animals with Attributes 2 (AwA2) dataset's images as ground truth images, inferring pose keypoints using the MMPose framework's model trained on the AP-10k dataset, however many of the AwA2 images were taken at angles, distances, and/or brightnesses which MMPose's model had difficulty making accurate inferences from. This resulted in roughly one third of generated pose estimates to be egregiously incorrect. Initial training using this dataset, even though it had over 30,000 images, would not converge due to this large proportion of incorrect pose estimates. Prompt text was not generated with BLIP in this example, instead using the defined animal classes in AwA2 as a generated prompt. For example, if there was a ground truth picture of a Zebra, it would have the prompt "zebra". A human-based method was used to screen out blatantly incorrect pose estimates, with the author manually selecting images using an automated program which would show images and keep or delete them with a single keystroke. Using this method the author was able to create a dataset of roughly 1,000 images of horses and oxen with a much higher percentage of accurate pose estimates, however the small size of this dataset proved to be insufficient for training a ControlNet. Choosing a small dataset, while normally very inefficient for training any network, has been proven to work robustly with ControlNet. The ControlNet paper itself shows an example of its canny edge model trained with differing dataset sizes, with datasets as small as 1,000 images reaching convergence without over-fitting, however with a noticeable decrease in generated image quality.[13]

4.2. Training Convergence

ControlNet's documentation in the train.md file on GitHub describes a phenomenon in training a ControlNet known as "sudden convergence".[1] This is where training will suddenly 'figure out' what a control input means. Instead of gradually showing closer results to the intended control output, such as animal poses slowly getting closer to the intended pose, ControlNet will start training generating images with seemingly no usage of the control input until it will suddenly start generating images which seem to use the control image. Once this convergence is reached, further training can be used to increase the quality of generated images. Sudden convergence for this model was reached around iteration 1200 of Epoch 7, after a total of 27,331 iterations at batch size 2. This convergence is shown in Figure 3.

5. Results

The trained model successfully uses animal pose as a control input for Stable Diffusion. The GitHub repository for this project can be found at https://github.com/abehonest/ControlNet_AnimalPose. A demo program to run inference using this model is included in this repository. A gallery of results is shown in Figures 5 to 16. Figures 15 and 16 show generation of animals not included in the AP-10k dataset: a platypus and a camel. For both instances this paper's model is able to generate an image matching the control pose, having never trained the ControlNet on these animals. Figure 9 shows an example of the model failing to generate a properly posed output, likely due to the rarity of both the pose and prompt animal (rhino) within the training dataset. Figures 13 and 14 show that unique style

can be added to images generated by this paper's model.

6. Discussion and Conclusion

This paper details the creation of a proof-of-concept implementation of an animal pose-based ControlNet. This proof-of-concept was trained for a relatively short period of time on a relatively small dataset yet was able to create a working ControlNet model for animal pose. To create a more robust model for this purpose, this paper recommends future work in creating larger animal pose datasets, with which new animal pose models can be trained on for longer periods of time. A graphic from the original ControlNet paper is shown in Figure 4, showing how inference quality varies with dataset size. For this new dataset it may suffice to use the additional roughly 50,000 non-annotated images within the AP-10k dataset and infer pose estimation on them, using a mechanical-turk-style system of manual filtering of poor pose estimates. The author of this paper attempted such a filtering on the AwA2 dataset, however found that he could only manually filter 1000 images per hour, indicating that a group of trained manual filterers may be needed to create such a new dataset in a reasonable period of time. This dataset will not be appropriate for training new pose estimation models on, however it should suffice for training a better animal pose ControlNet using inferred poses as 'ground truths'. It is expected that training this new model would show a similar sudden convergence time to this paper's model, allowing future researchers to know within roughly 60,000 iterations (at batch size 1) if their model will converge or if further data filtering is needed. For each of this paper's listed failed attempts at generating a model, convergence was not reached within roughly 150,000 iterations. Future iterations of this model may also be ported to ControlLoRA to lower model size.

This paper also touches on several different datasets available for animal pose estimation, such as the Horse-10 dataset, which uses frames from several videos of horses. Animal pose datasets using continuous frames from video should not be used for this application, at least as the bulk of data, in order to minimize the chance of over-fitting.

7. Figures

Figure 1. Generating Quadrupeds using a Bipedal Pose in the Openpose Model

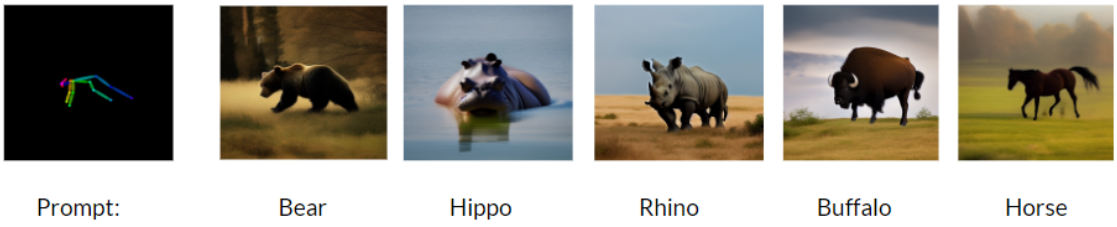

Figure 2. Generating Quadrupeds using a Quadrupedal Pose in the Openpose Model

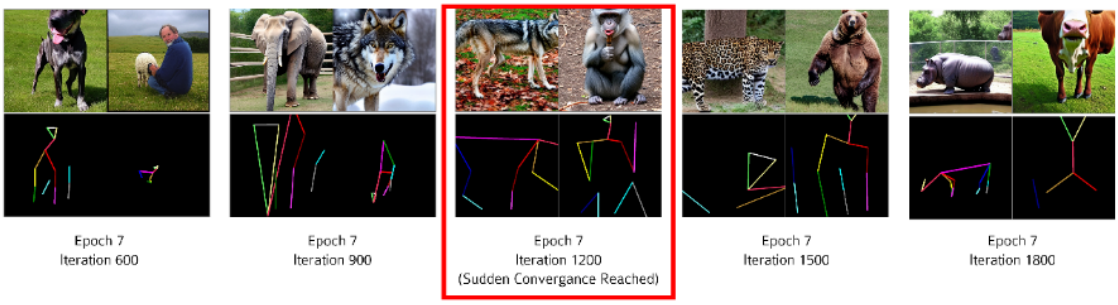

Figure 3. Reaching Sudden Convergence at Iteration 1200 of Epoch 7

Figure 4. How the Canny-edge-based ControlNet trained on different experimental settings with various dataset size.[13]

Figure 5. Generated image with prompt "A deer walking on a grassy field"

Figure 6. Generated image with prompt "A horse walking in a snowy forest"

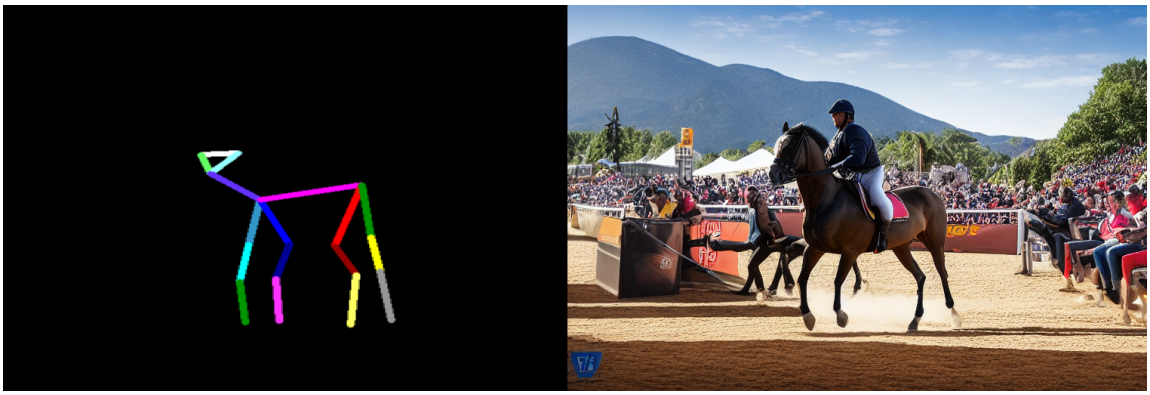

Figure 7. Generated image with prompt "A man riding a horse"

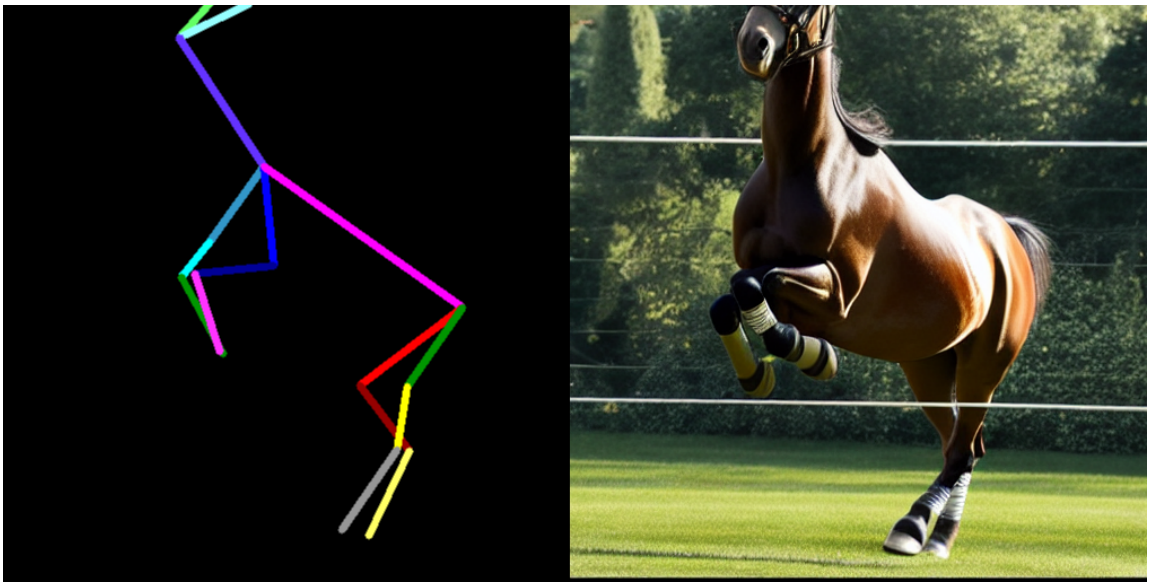

Figure 8. Generated image with prompt "A horse"

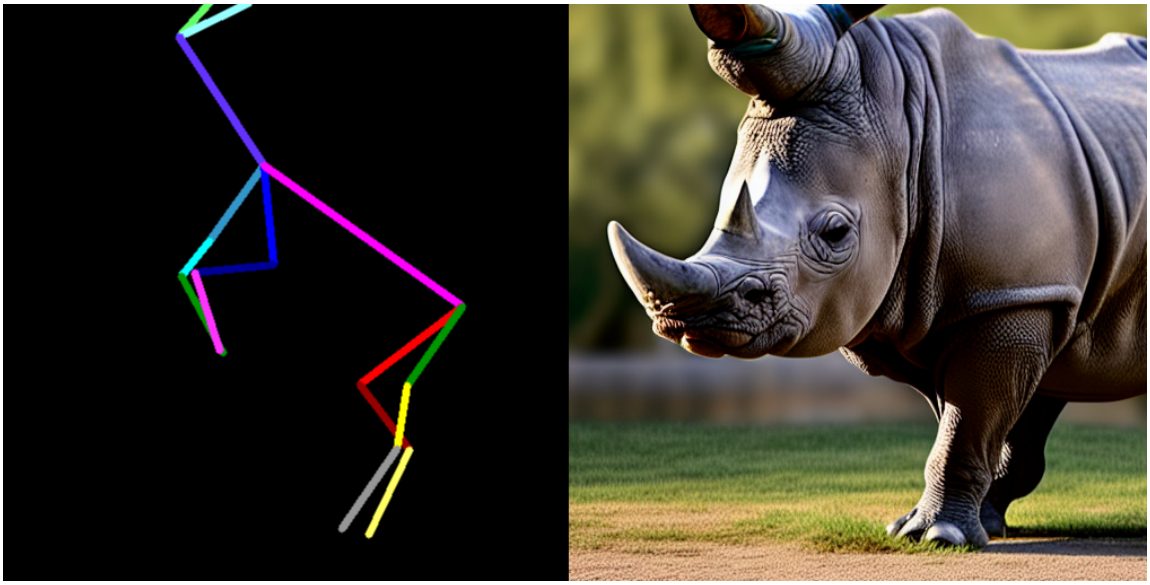

Figure 9. Generated image with prompt "A rhino with a red horn"

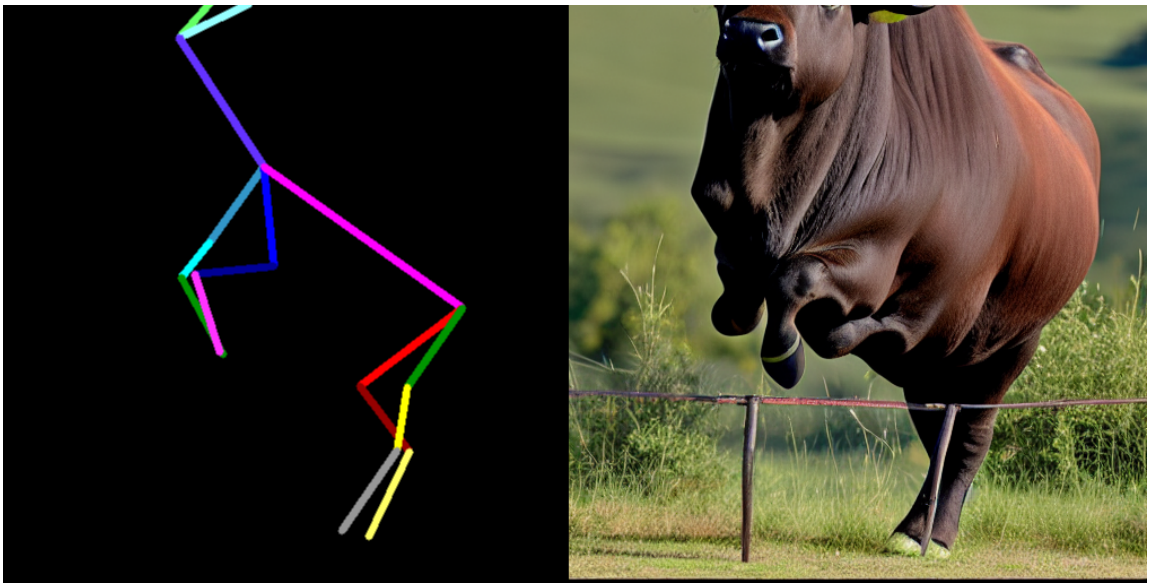

Figure 10. Generated image with prompt "An ox"

Figure 11. Generated image with prompt "A cow jumping over a wall of fire"

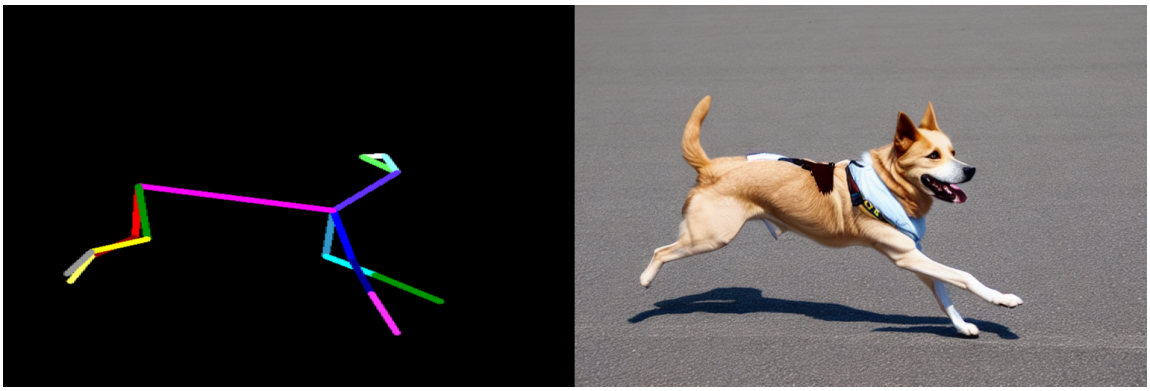

Figure 12. Generated image with prompt "A running dog"

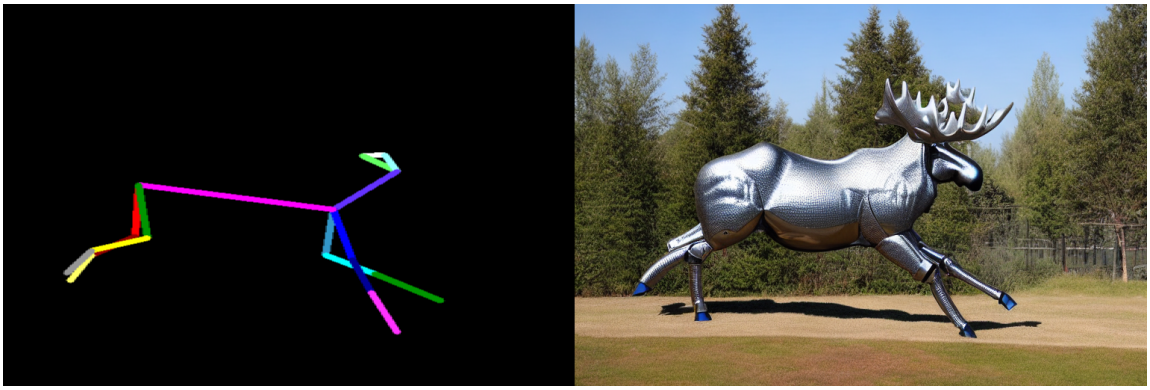

Figure 13. Generated image with prompt "A robotic metallic moose"

Figure 14. Generated image with prompt "A hot pink cheetah"

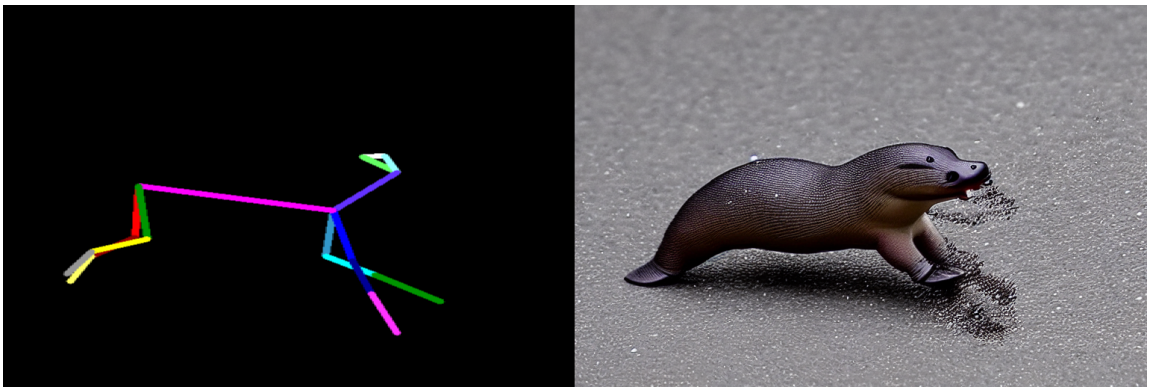

Figure 15. Generated image with prompt "A platypus running"

Figure 16. Generated image with prompt "A camel running"

References

[1] Stable Diffusion, May 2023. original-date: 2022-08-10T14:36:44Z.

[2] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. Cross-Domain Adaptation for Animal Pose Estimation, Aug. 2019. arXiv:1908.05806 [cs] version: 2.

[3] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields, May 2019. arXiv:1812.08008 [cs].

[4] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models, Oct. 2021. arXiv:2106.09685 [cs].

[5] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. OpenPifPaf: Composite Fields for Semantic Keypoint Detection and Spatio-Temporal Association, Sept. 2021. arXiv:2103.02440 [cs].

[6] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation, Feb. 2022. arXiv:2201.12086 [cs].

[7] Alexander Mathis, Thomas Biasi, Steffen Schneider, Mert Yüksekgönül, Byron Rogers, Matthias Bethge, and Mackenzie W. Mathis. Pretraining boosts out-of-domain robustness for pose estimation, Nov. 2020. arXiv:1909.11229 [cs].

[8] MMPose Contributors. OpenMMLab Pose Estimation Toolbox and Benchmark, Aug. 2020. original-date: 2020-07-08T06:02:55Z.

[9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models, Apr. 2022. arXiv:2112.10752 [cs].

[10] Hecong Wu. ControlLoRA: A Light Neural Network To Control Stable Diffusion Spatial Information, Feb. 2023. original-date: 2023-02-18T09:12:15Z.

[11] Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-Shot Learning – A Comprehensive Evaluation of the Good, the Bad and the Ugly, Sept. 2020. arXiv:1707.00600 [cs].

[12] Hang Yu, Yufei Xu, Jing Zhang, Wei Zhao, Ziyu Guan, and Dacheng Tao. AP-10K: A Benchmark for Animal Pose Estimation in the Wild, Nov. 2021. arXiv:2108.12617 [cs].

[13] Lvmin Zhang and Maneesh Agrawala. Adding Conditional Control to Text-to-Image Diffusion Models, Feb. 2023. arXiv:2302.05543 [cs].