Human Pose Based Video Generation with Dual ControlNet-Enhanced Stable Diffusion

David Wang Dijkstra Liu Joshua Tang

Abstract

This project introduces a novel framework for generating video sequences from motion capture video streams harness the capabilities of generative diffusion models, particularly based on the Stable Diffusion [3] model and enhanced by ControlNet [6]. Our method leverages the unique capabilities of selected ControlNet models to maintain frame-to-frame consistency while adapting to dynamic human poses derived from motion-captured video data. Specifically, we proposed a approach leveraging dual-layer controlNet(Pose and Reference-Only control) to ensure each video frame is generated with precise adherence to the captured human poses while maintaining consistency in non-pose-related aspects across frames. We present a comprehensive work starting from the creation of a specialized dataset using OpenPose [1] and annotations of the data using OpenAI's API to train a human-pose ControlNet. The workflow for generating the video is supported by extensive experiments over parameters. Our results demonstrate the effectiveness of our method in generating coherent and visually consistent video sequences, showcasing the potential of integrating pose estimation with for creative and practical applications in digital media and animations.

1. Introduction

In recent years generative AI has witnessed remarkable growth with innovative applications being used in various industries. Among these advancements, image generation has been a popular application that has been finding use cases among professionals and amateurs. The next step in generative AI is naturally video generation and OpenAI has recently released a model called Sora with its abilities to transform text prompts into video sequences. Despite the promising examples we have been shown, text input is often limited in their ability to describe intricate details which is best represented by the saying that a picture is worth a thousand words. Recognizing this limitation, our project focuses on creating a system that allows the user increased control over video generation, specifically through the manipulation of human poses within the video.

To create the video we wanted to use a diffusion model to generate a series of images that could be put together to make a video. The core part of our approach is to use ControlNet which is a model that controls the output image from diffusion models using an additional input image with the text input. ControlNet can take a wide variety of additional input images to influence the output including canny edge images or even user drawings. Since we are focused on controlling the human pose of the output images, we decided to use OpenPose inputs which is a pose estimation system.

2. Related Work

2.1. OpenPose [1]

OpenPose is a program developed by the Carnegie Mellon Perceptual Computing Lab that represented a significant advancement in multi-person pose estimation. Unlike other methods, OpenPose uses part affinity fields that allows it to do multi-person pose estimation in real-time at a higher speed and accuracy than previous methods. Since the development of OpenPose, its applications have been widespread, from animation and gaming to more complex research topics. Due to the speed and accuracy of OpenPose, we decided to use OpenPose for our project as the input to ControlNet.

2.2. Stable-Diffusion [3]

The core part of our project is to generate a series of images to create a video. To do that we decided to use Stable Diffusion which is a cutting-edge text-to-image diffusion model that has been very popular in the field of generative AI. Stable Diffusion was developed by researchers at Ludwig Maximilian University and uses a latent diffusion process. They combined that process with a Clip-based guidance system to transform text prompts to an embedding space that will be used to influence the diffusion process. Its open-source nature has helped with creating a community of developers who continue to expand on its applications. This includes ControlNet which is a model that is pivotal to our project.

2.3. ControlNet

With the Stable-Diffusion as the base model for images generation, we implemented ControlNet [6] to achieve human pose-based generation and cross frame consistency. ControlNet is an add-on to the diffusion model that is designed to fine-tune the control over the Stable Diffusion model.

Typically, stable diffusion accept the textual prompt as the only control over the generation process. With the ControlNet, we can add various control without re-training a entire stable diffusion that based on other kind of control. ControlNet enhances the image generation process by expanding the original encoder of Stable Diffusion. Specifically, ControlNet replicates the structure and parameters of the original encoder blocks and the middle block of Stable Diffusion, utilizing them as a deep, robust, and powerful backbone network to learn diverse control conditions for image generation(as shown in Fig. 1). By adding additional trainable layers, ControlNet learns to influence the image synthesis, allowing for more nuanced guidance. Notably, the process does not require the retraining of the Stable Diffusion model which usually trained on significantly larger dataset(about 5 billion images with LAION-5B dataset [3]), meaning that despite the added layers, the required GPU memory for generating does not significantly exceed that of the original model. More importantly, training a controlNet only requires a really small dataset compare with the one used to trained the stable diffusion, allowing users to customize their controlNets. This approach makes training ControlNet computationally efficient.

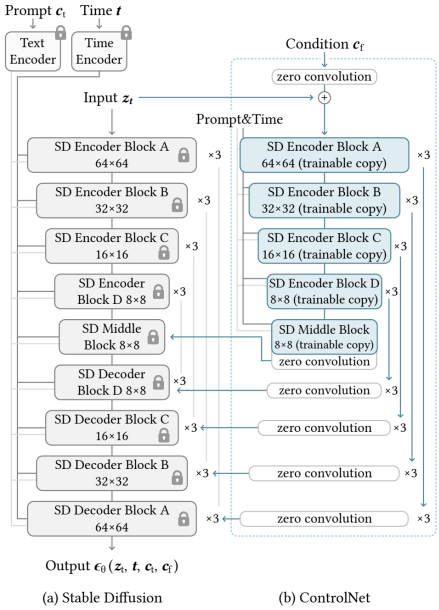

Figure 1. ControlNet's Structure [6]

3. Approach

Leveraging human pose images as ControlNet inputs to guide a diffusion model, we propose a method to generate video sequences by processing a series of human pose images derived from motion-captured video. Each pose image is used to generate a corresponding video frame, which are then concatenated to form a cohesive video. The main challenge here is to maintain the consistency between each frame but have the human poses changed. We proposed to use the Reference-Only and Pose ControlNet together to generate video from a sequence of pose images while ensure the frame to frame consistency. Reference-only control does not need to be trained since it is achieved by connecting a reference image directly to the attention layers of the stable diffusion without any encoding process, so we only need to train a human pose controlNet. Therefore, our approach includes following steps: 1. building dataset utilizing OpenPose and OpenAI's API 2. training the pose ControlNet 3.generating the video.

3.1. Build dataset

To train a controlNet, there are three input parts needed. Firstly, a set of target images (Fig. 2) that is used as the ground truth to compute the training loss.

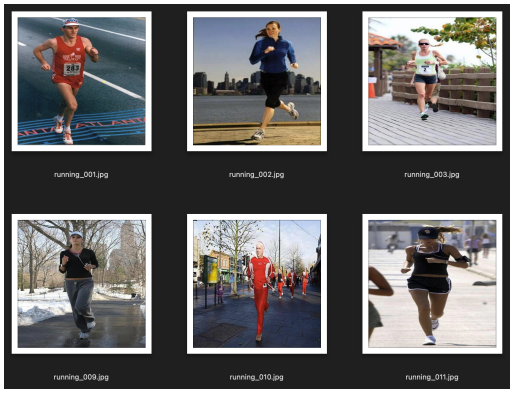

Figure 2. Target Images

Second, a set of source images (Fig. 3) that is generated from its corresponding target, which is our customized control, and here in our case we will use the human poses images.

Lastly, a list of prompts (Fig. 4) of the target image is needed for guiding the stable diffusion model to generate the images.

- **Data Collection:** We used Stanford 40 Actions dataset [4] as the foundation for building our own dataset that is ready for training a pose ControlNet. This dataset contains about 10K images of people doing different things. However, this dataset does not

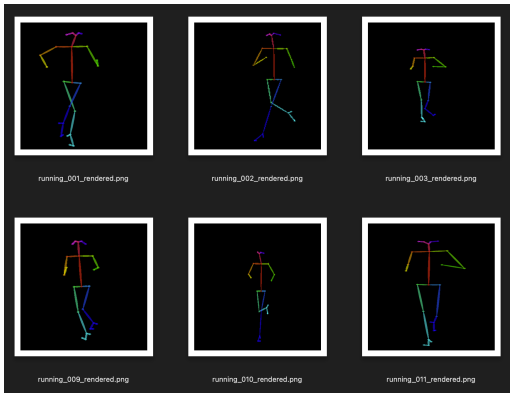

Figure 3. Source Images


```
{
    "source": "source/running_001_rendered.jpg",
    "target": "target/running_001.jpg",
    "prompt": "A runner is captured mid-stride on a marked section of road
        denoting \"ATLANTA.\" The athlete is dressed in a red singlet and
        shorts, with a white cap and matching red and white running shoes. They
        sport the number \"283\" on their race bib, which also includes the
        Atlanta Committee for the Olympic Games logo, indicating this may be
        during an Olympic trials event. The focus and determination on the
        runner's face suggest a high level of competitiveness in what appears
        to be a long-distance running event."
},
{
    "source": "source/running_002_rendered.jpg",
    "target": "target/running_002.jpg",
    "prompt": "A woman is jogging along a waterfront boardwalk with a backdrop
        of a city skyline under a clear blue sky. She is dressed in athletic
        attire: a blue long-sleeve jacket, black pants, and running shoes, and
        her hair is tied back as she maintains a steady gaze forward,
        indicating focus on her exercise."
},
{
    "source": "source/running_003_rendered.jpg",
    "target": "target/running_003.jpg",
    "prompt": "A female runner is captured mid-stride as she competes in a
        race. She is wearing a green tank top, black shorts, and green running
        shoes with white socks. Her race number, \"2,\" is pinned to the front
        of her top. She sports sunglasses and seems to be intensely focused on
        the race. The runner is on a wooden boardwalk lined with a wooden
        fence, and there are shrubs and trees in the background suggesting the
        race might be taking place in a park or along a scenic route. The
        photograph's depth of field is shallow, with the runner in sharp focus
        and the background blurred, emphasizing her motion and the energy of
        the moment."
},
```


Figure 4. Prompts

include the prompt nor has the human pose extracted, therefore we need to use the following tool to augment the data.

- **Openpose [1]:** In our project, OpenPose plays the role as the primary source of input for ControlNet. It helps facilitate the control of human poses within the generated video frames. We used OpenPose on various videos of ourselves and other videos to create the series of OpenPose images needed to produce a video. We also used OpenPose to create our datasets by running OpenPose on the images from Stanford 40 Actions datasets.

- **OpenAI [2]:** We utilized OpenAI's GPT-4 vision API to help us annotate the image data by letting it describe all images in the dataset where we use the output as the prompt(source code for implementing this in the zip file).

Since the dataset contained so many images, we had to write a script that created a json file that listed the location of all the images and prompts.

3.2. Train

After we have the dataset ready, we started the training process. ControlNet official github repo [5] provides the baseline for training a controlNet. In order to have the environment, dataloader, and parameters setup correctly, we ran a simple example of controlNet with a pre-build dataset that is ready for training. In this simple demo, we trained a controlNet that controls Stable Diffusion to fill color in circles with designated size and location(as illustrated by Fig. 5). After making sure that the baseline was setup correctly and did not crash, we feed our data to the baseline with a similar training process as Fig. 5. Since the baseline has been proved working by running the demo, we can debug our dataset format accordingly to fix any mistakes from building our dataset.

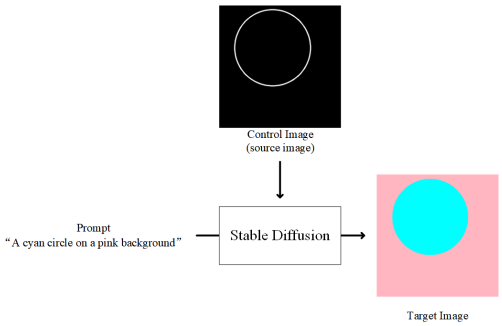

Figure 5. ControlNet to fill color in circles [5]

3.3. Video Generation

The process for generating videos involves several stages, each leveraging advanced tools from video processing to controlled image synthesis. The following steps outline the workflow(see visual illustration in Fig. 6):

1. **Video Acquisition:** The initial step in the video generation process involves capturing or acquiring a base video. This video serves as the raw material on which further processing is applied. Depending on the intended outcome, this might involve filming real-world scenes, capturing human actors, or creating animations. In our case, we select a video from YouTube https://www.youtube.com/watch?v=Bsll7xLEWh8 as the raw material.

2. **Frame Extraction and Processing:** Once the video is acquired, it is typically broken down into individual frames. Using a frame rate of 20fps (frames per second) maintains the detail and smoothness of the video, making each moment more distinct. Additionally, replacing the video background with a green screen can

significantly aid in pose extraction, as it provides a uniform background that simplifies the detection of figures and movements, though this step is optional.

3. **Pose Estimation with OpenPose:** For videos involving human actors or specific objects that require tracking, pose estimation and object recognition tools are applied. OpenPose is used to analyze each frames to detect human poses. The result from OpenPose is crucial for ensuring that the generated content adherence to designated human pose.

4. **Prompt Creation and Refinement with OpenAI's API:** GPT-4 Vision model for image recognition allows for the generation of more detailed and descriptive prompts. This model can analyze images to identify key elements and attributes, thereby producing text prompts that better capture the nuances and specifics of the visuals, leading to more accurate textual based image generation.

5. **Image Generation with Stable Diffusion + Dual-layer ControlNet:** Generate a single, high-quality image using the refined prompt with a Stable Diffusion model. This image will be used as the benchmark. Then, we used the poses extracted by OpenPose as the first layer of control and use this benchmark image as the second layer of control. The first layer we used is a pre-trained pose controlNet that is much more accurate than the one we trained. The second layer is a reference-only control. With this dual-layer control structure, we generated a series of images where each frame corresponds to the pose of the astronaut at each point in time, based on the reference-only ControlNet that used as the benchmarks. The principle of 'reference only' is to replace the input feature map to the decoder of the Diffusion model with the feature map of the benchmark image, so that a similar style can be maintained to ensure across frame consistency. By adjusting the control weight and various parameters of this two layers, we would be able to get consistent and coherent images.

6. **Compile Generated Frames into Video:** Take the generated images for each pose and assemble them back into a video sequence, ensuring the frame rate matches the original video for fluid motion. Review the generated video to ensure continuity and visual quality. Adjust timings or regenerate specific frames if necessary to improve the seamless integration of the dance movements.

7. **Finalizing and Exporting:** The final step involves detailed editing tasks such as adjusting color balance, aligning audio with visual cues, and ensuring seamless transitions between frames. Critical attention is paid to the continuity and coherence of the visual narrative to maintain the story's flow. After these adjustments, frames are meticulously merged into a video sequence using images editing software Fiji.

4. Results

4.1. Training

We start from training a simple controlNet that is not directly related to our purpose but helped us to understand the training process and validate our own dataset. After this, we start to train a more complicated pose controlNet.

- **Simple Circle ControlNet Training:** Training this controlNet used a dataset consists of 50K images of synthetic circles, where 20% of images are draw from the dataset and used for testing. This training process adjusted the batch size to 4 based on the available GPU memory, with the u-net architecture where the parameters are set as default. After the check point at 2700 steps, the results shows some unwanted artifacts and textures that are inherited from the Stable Diffusion(Fig. 7). After the 7800 steps, we saw a reduction in unwanted texture, however, the alignment of the circle is still off(Fig. 8). After 11700 step, the result shows a convergence to the contour of the circles (Fig. 9).

- **Pose ControlNet Training:** Upon observing the result of training a simple controlNet, we've validate our dataloader and trainer are correctly setup. Then, we train the pose control model with a subset of 2K images of all 10K images, where we discovered some defects with our dataset that crashed the training code. After fixing the dataset by reformatting the .json file for the data loader, we were able to start the training. We are not using all 10K images because we were not able to annotate all images due to the rate limit of OpenAI's API. However, 2K images can give some visible effects to the Stable Diffusion Model thanks to the great scalability of the ControlNet structure. It is obvious that the pose control is much more complicated than filling the circles, so that it will take much more steps to converge. After the check point at 4632 steps in the 12th epoch, we still cannot see any correlation between the source and the generated image(Fig. 10). The testing result start to show some convergence around 14968 steps at 38th epochs, but the alignment to the pose is not stable across different testing images Fig. 11. At around 20458 steps at 53th epoch, the result shows a more consistent convergence as shown in the the Fig. 12. Due to the limited

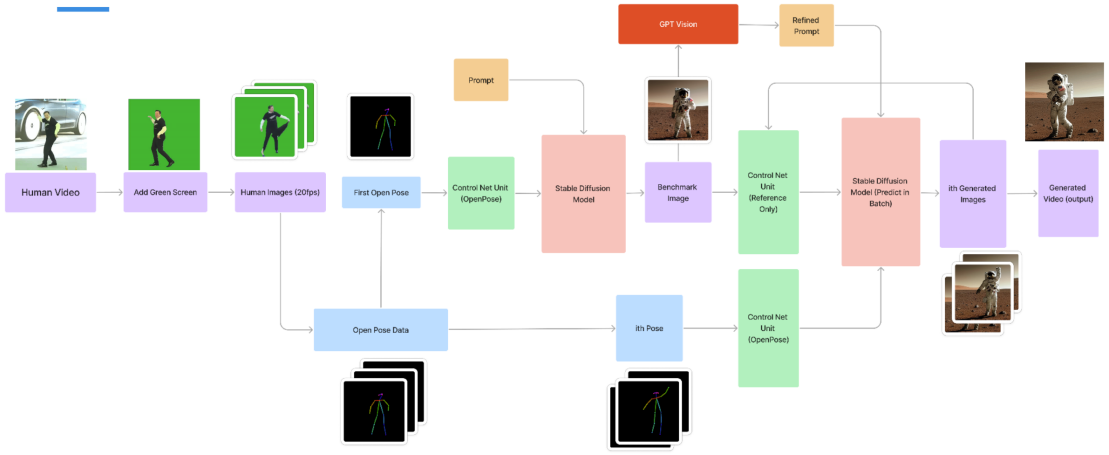

Figure 6. Work Flow for Video Generating Process

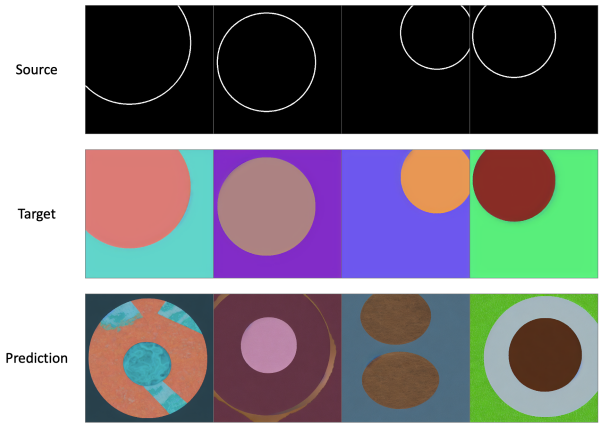

Figure 7. Testing Result of 2700 steps

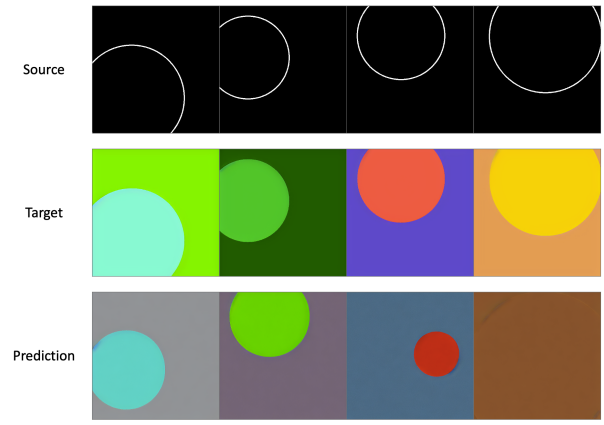

Figure 8. Testing Result of 7800 steps: no more noisy textures, but circles are not well aligned

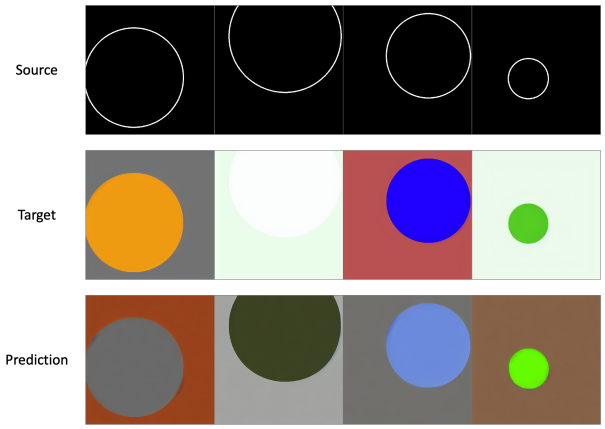

Figure 9. Testing Result of 11700 steps: decent contours' alignment

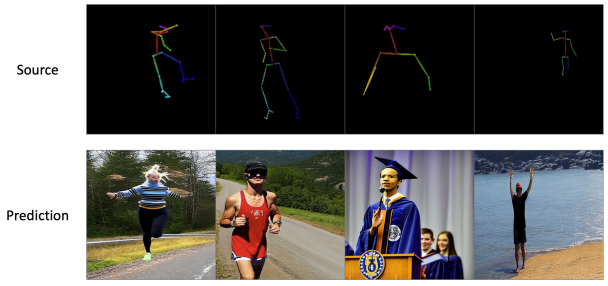

Figure 10. Testing Result of 4632 steps: training loss = 0.187

size of the dataset, the result from training the pose control is obviously not satisfactory. Therefore, for the remaining video generating process, we will used a pre-trained pose controlNet that was trained on a larger

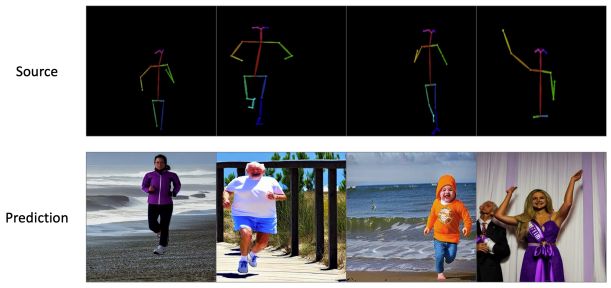

Figure 11. Testing Result of 14968 steps: training loss = 0.156

Figure 12. Testing Result of 20458 steps: training loss = 0.129

dataset.

4.2. Video Generation

In the video generation process, the design of the controlNet and the adjustment of the hyperparameters for the Stable Diffusion are important. We found that using reference control in controlNet in addition to the pose control can maintain the consistency in style and scene with the original image. Compare the following result of using and not using reference-only control we could see why we need this second layer of control. As shown in Fig. 13, using the identical random seed and prompt but without the reference-only control, the style does not looks alike and it does not make sense to have two differently styled images in one coherent video.

In contrast, by using the first generated image as the reference, two images from two different generated process have the same background and even the rocks on the ground are identical.(as shown in Fig. 14).

Additionally, the weight assigned to openPose Control also needs to be increased. A insufficient control of pose controlNet will result in defects looks like Fig. 15, which is generated with the same pose as shown in Fig. 14 but the weight of pose control is lower.

In contrast, the prompt's weight during video generation is not as crucial, as it serves more as a reference rather than a decisive factor. Overemphasizing the prompt can lead to significant differences between generated video frames, resulting in poor quality. To further elaborate,

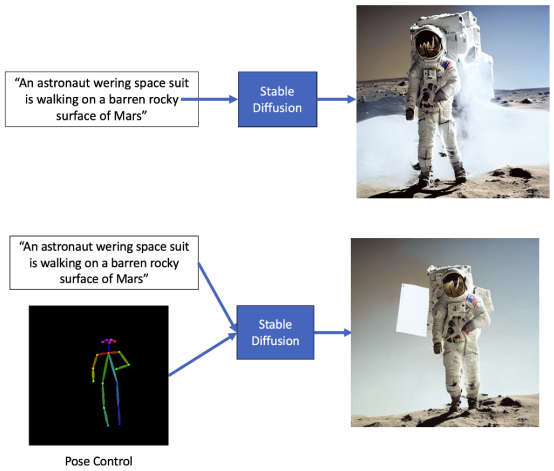

Figure 13. Two generated images from the same prompt without Reference-only control.

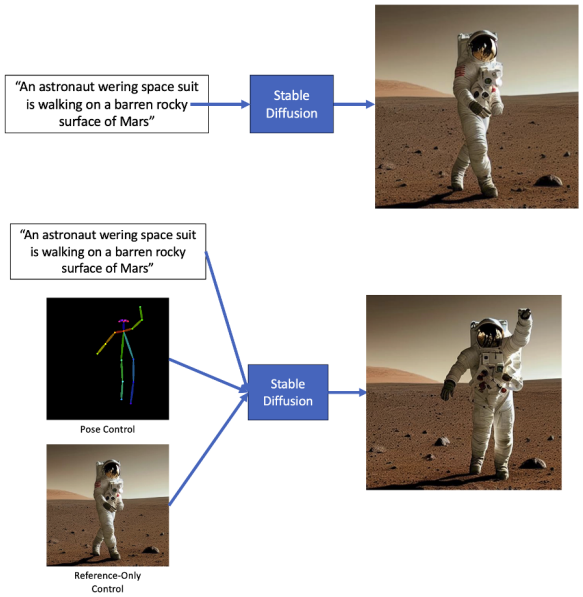

Figure 14. Two generated images from the same prompt with Reference-only control.

our results indicate that fine-tuning the control mechanisms within the video generation pipeline allows for more precise manipulation of both the stylistic elements and the dynamic aspects of the video content. By adjusting these control weights, we are able to achieve a higher consistency across video frames, which is critical for maintaining the narrative flow and visual coherence in generated videos. This approach not only improves the aesthetic appeal of the videos but also enhances the overall viewer ex-

Figure 15. Example of issue arises from low pose control weight

perience by providing smoother transitions and more predictable movements, crucial for applications in digital storytelling and interactive media. Our final generative video can be found here: https://www.youtube.com/watch?v=GlM38Tqtq4c

5. Discussion and Conclusions

Upon reviewing the training results of the ControlNet, we identified several areas for improvement. Firstly, expanding our dataset could significantly enhance our results. For instance, training the circle control with a larger dataset of 50,000 images yielded satisfactory outcomes. In contrast, the smaller dataset used for the more complex pose control did not allow for accurate control. Secondly, the our pose ControlNet is designed for images with a single person, but our original dataset included many images with multiple people. Although we restricted the number of individuals in the pose images during generation, the presence of multiple people in the background introduced too much variability. To address this, we can preprocess the dataset to exclude images featuring multiple people. Lastly, the training of the ControlNet does not require detailed prompts; in fact, simpler prompts could improve its ability to recognize the semantics of the input conditioning images, substituting for detailed textual prompts. Previously, we annotated each image with approximately three full sentences, which proved to be overly detailed. We can reduce the complexity of the textual prompts by either employing prompt engineering to generate shorter responses from GPT-4 or by substituting some text prompts with empty strings. Throughout the training process of the ControlNet, we have deepened our understanding of how diffusion models and attention mechanisms function and have gained valuable insights into enhancing the ControlNet's training.

After implementing the reference only and OpenPose control, the style of the video was essentially consistent, and the movements were based on the original video's actions. However, there are still differences in frames' details, such as changes in the relative position of the background, and inconsistencies in facial expressions and finer details of the characters. In subsequent training, controlNets similar to depth can be used to maintain the constancy of the background object positions. Specifically, the relative spatial especially depth information cannot only be captured by semantic style, we could use a third layer of controlNet which is depth-map to maintain the spatial consistency. Additionally, using a more refined sampling method would enhance the precision of image generation. Moreover, upgrading from the basic OpenPose to OpenPose Full, which can capture facial expressions and hand movements, will also improve the quality of the generated videos.

6. Statement of Individual Contribution

6.1. David Wang

Reviewed literature to devise a manageable approach. Then tested and compared various control types, selecting a combination of pose control and reference-only control for our generating pipeline. Annotated images utilizing OpenAI's API. Refine the dataset and fixing unwanted data by modified data loader code(part of training code). Finally, tuned parameters and trained the Pose ControlNet(tuned training code). Handled tasks distribution. Helped with all the slides and reports.

6.2. Dijkstra

Tested and compare various controlNet types with SD-webUI, ultimately selecting a combination of pose control and reference-only control for our generating pipeline. Annotated images utilizing OpenAI's API(wrote annotator code). In video generation, optimize and complete the video generation pipeline(build the workflow chart). Deploy and apply the SD-webUI to generate final video result. Helped with all the slides and reports.

6.3. Joshua Tang

Helped search for datasets that contained images that would be ideal for our dataset. Researched background of Stable Diffusion and ControlNet. Also looked into OpenPose and how to use it for our datasets and projects. Finally, helped put the dataset for our ControlNet training together and helped with all the slides and reports.

7. External Resources Used

7.1. OpenPose

To use OpenPose we decided to use the .exe version since we didn't need to edit any of this code. We followed the tutorial and ran OpenPose on the Stanford dataset with certain parameters like only detecting 1 person and deleting the background for ControlNet. https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/doc/01_demo.md

7.2. OpenAI's API

We used OpenAI's API to annotate the images. Sepcifically, we used the "gpt-4-vision-preview" mdoel. By posting the request contains a textual prompt and an image to OpenAI's sever, we are able to extract the desired response. In order to minimum the task of processing the respond, we did some prompt engineering to make the model give us more concise responses. We applied this process to all images in the folder. Documentation of this API is here: https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4

7.3. ControlNet

This official implementation of ControlNet provides details of how to train a customized controlNet and apply different controlNets. Here we used this as the baseline to train our own pose control. We firstly follows the environment file to setup the training environment using pytorch lightning, Then, we adopted the main training code and tune the parameters to make it better fit onto our dataset and GPU. The circle controlNet is a demo for training from this github repository. Link to this repo: https://github.com/lllyasviel/ControlNet/tree/main

7.4. Stable Diffusion-webui

Stable Diffusion-webui is an open-source user interface designed to facilitate the use of the Stable Diffusion model for image and video generation tasks. This tool allows users to interactively control and modify the parameters of the Stable Diffusion model, enabling the creation of customized visual content. In our project, we utilized Stable Diffusion-webui to generate and refine the visual components of our video generation pipeline. Specifically, we leveraged its capabilities to adjust the style and detail of generated images, ensuring that they align with the desired thematic and stylistic guidelines of our video content. The intuitive interface of SD-webui significantly streamlined our workflow, allowing for real-time adjustments and previews, which was crucial for rapid prototyping and iterative development. The source code and further documentation for Stable Diffusion-webui can be accessed at https://github.com/AUTOMATIC1111/stable-diffusion-webui.

References

[1] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. volume abs/1812.08008, 2018. 1, 3

[2] OpenAI. Openai api. https://platform.openai.com/docs/models/overview, 2024. 3

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2

[4] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *ICCV*, pages 6–13, 2011. 2

[5] Lvmin Zhang. Controlnet. https://github.com/lllyasviel/ControlNet, 2023. Accessed: 2024-02-28. 3

[6] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 1, 2