**workshops[1-4]_RC11**

2023 - 2024

Joris Putteneers

# Part I
# Workshops

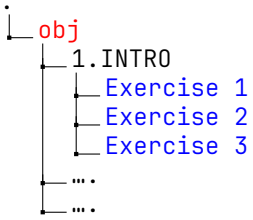
## 1 Workshop: Houdini Fundamentals

### 1.1 Overview

In the first workshop, we will cover the Houdini fundamentals. The example file can be downloaded from github.

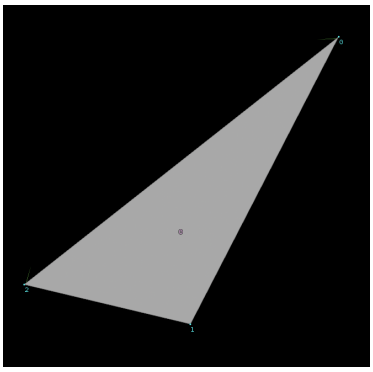
#### 1.1.1 Houdini Introduction

1. Viewport navigation.
2. Object and geometry contexts.
3. What are points, primitives, vertices.
4. What are the most common data types.
5. Exercise 1: our first geometry.
6. Point and primitive attributes.
7. Attribute promotions.
8. Using attributes to drive geometric operations.
9. What is proceduralism.

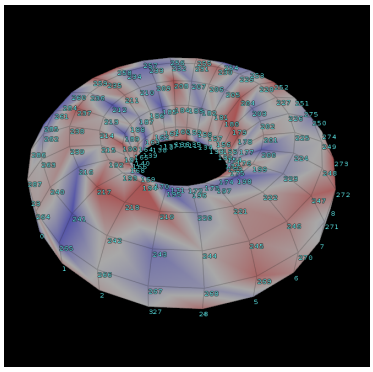



- ■ Current exercises
- ■ root node

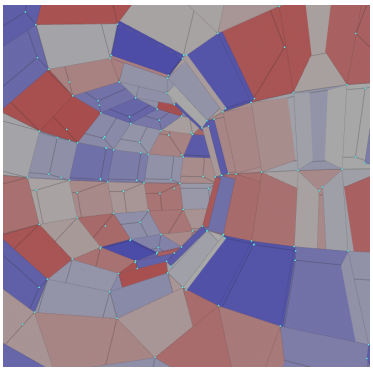
Figure 1: Tree structure of Houdini exercises



(a) detail: exercise 1



(b) detail: exercise 2



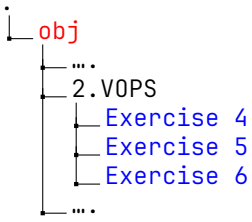
(c) detail: exercise 3

Figure 2: Houdini introduction: exercise previews

At the end of this session, you will be able to create a basic procedural node setup, understand the different contexts, as well as understand the basic geometric model Houdini uses.

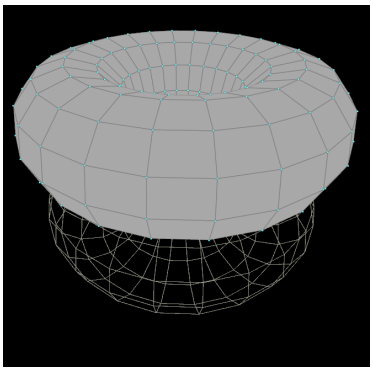
### 1.1.2 VOPS and VEX

1. VOPS and VEX introduction.
2. Geometry manipulation with VOPS and VEX.
3. Global and local primitive and point attributes.
4. Exercise 1: Moving geometry with VOPS and VEX.
5. Exercise 2: Make a procedural staircase.
6. Noises in VOPS.
7. Exercise 3: Position manipulation with noises.

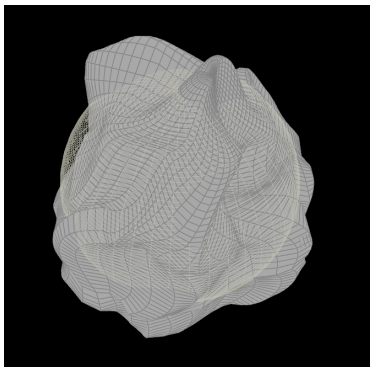



- ■ Current exercises
- ■ root node

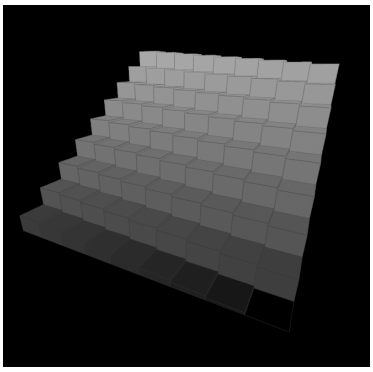
Figure 3: Tree structure of Houdini exercises



(a) detail: exercise 4



(b) detail: exercise 5



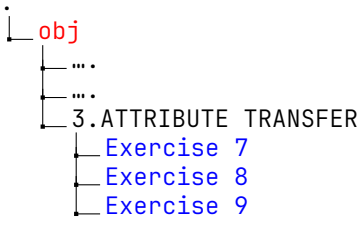
(c) detail: exercise 6

Figure 4: VOPS and VEX: exercise previews

At the end of this exercise you will be able to procedurally drive attributes with VOPS and VEX.

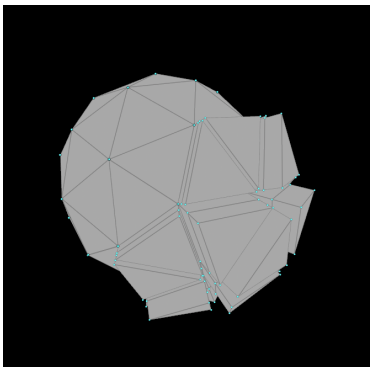
### 1.1.3 Controlling Attributes

1. Attribute interpolations.
2. Parameter referencing.
3. Exercise 3: Linear attribute interpolations for heat maps.
4. Rendering our scene with OpenGL.

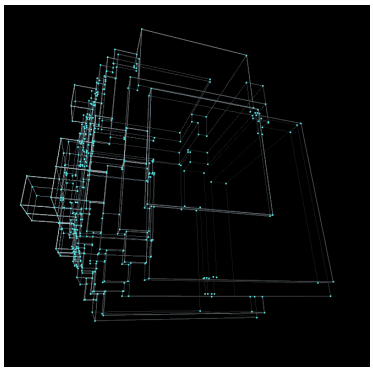



- ■ Current exercises
- ■ root node

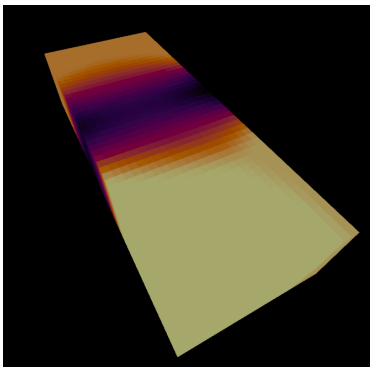
Figure 5: Tree structure of Houdini exercises



(a) detail: exercise 7



(b) detail: exercise 8



(c) detail: exercise 9

Figure 6: controlling attributes: exercise previews

At the end of this exercice you will be able to use attributes in a controlled way.

---

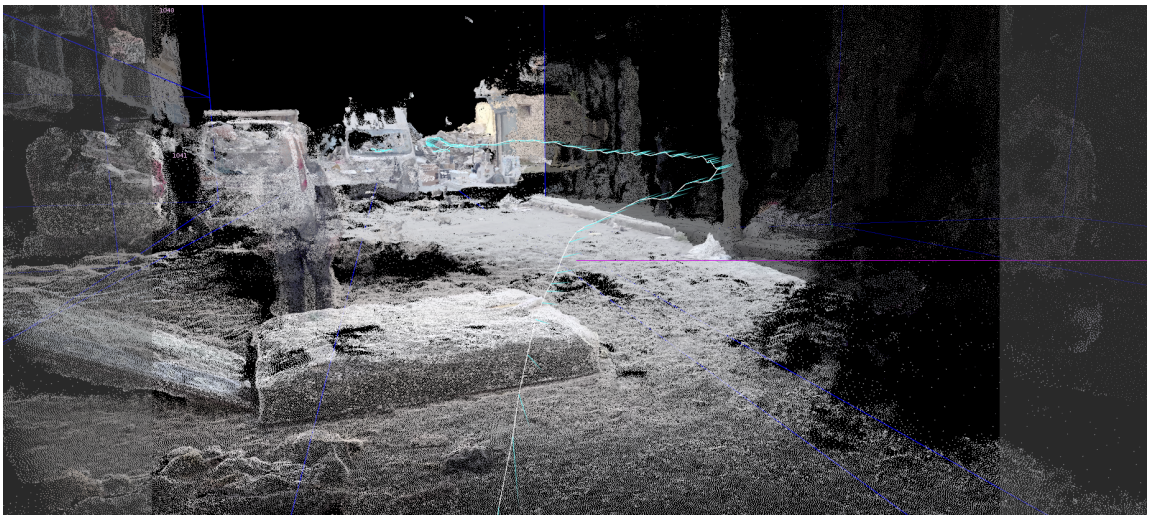
## 2 Workshop: mapping the 2023 Turkey-Syria earthquakes

### 2.1 Overview

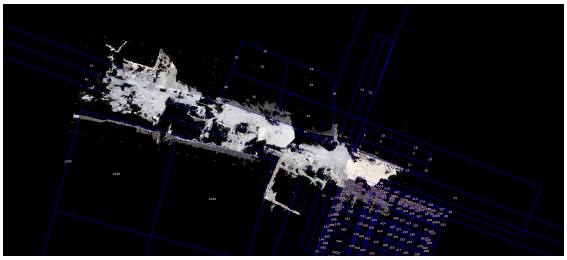
On February 6, 2023, a magnitude 7.8 earthquake struck southern and central Turkey as well as northern and western Syria. To assess the impact of the damage, we can try to reconstruct certain areas.

The example file can be downloaded from github.

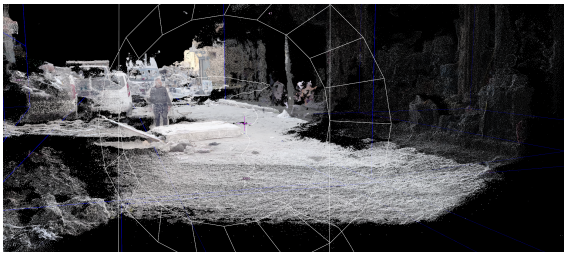
1. Scrape for footage on social media with PyTube.
2. Extract frames from the footage with openCV.
3. Use photogrammetry tools such as colmap, Gaussian splatting and realitycapture to reconstruct the 3D model.
4. Reverse engineer the location of the scene.
5. Reconstruct the original model using Google Maps and FSpy.
6. overlay the original building on the pointcloud and Reconstruct the camera path.
7. Render out an OpenGL frame sequence with Houdini and compose with FFmpeg.



(a) camera fragment



(b) Bird's Eye View



(c) Detail: Road Map

Figure 7: trajectory and 3d scan

---

# 3 Workshop: Visualizing GPX trajectories with OSM data and metadata tagged Images

## 3.1 Overview

With the Houdini fundamentals in mind, in the third workshop, we will visualize a GPS route combined with OSM data and images in Houdini.

The example file can be downloaded from github.

The procedure is as follows:

- We will begin by going on a walk in your current city (minimum 20 minutes). Ensure that the location on your phone is turned on.
- Take pictures of interesting spaces during this walk (minimum of 10 images).
- Taking multiple modes of transportation, such as a bus or tram, car, or bike, might provide additional interesting information.

### 3.1.1 Validating Data

1. Extract your .kml file from Google Maps.
2. Convert your .kml file to .gpx, a more easily readable data format. Use this website.
3. Get the images on your machine in an uncompressed format. An easy way is to upload them to Google Drive or GitHub.
4. Check if your images contain the correct EXIF metadata.

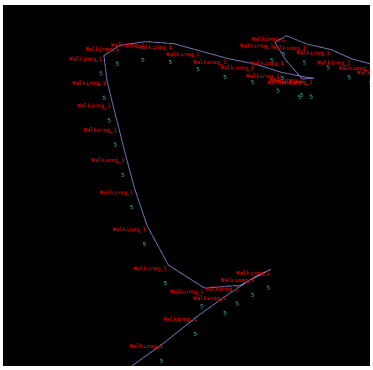
5. Make sure your .gpx file contains waypoints <wpt></wpt> and tracks <trk></trk>.

### 3.1.2 Visualizing Data

When we have successfully validated our data, we can start visualizing this trajectory in Houdini.

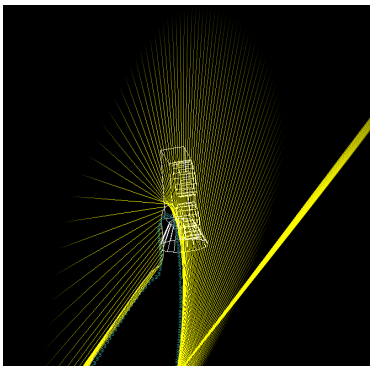
#### 3.1.2.1 Reconstruct .gpx Data:

1. Extract data from .gpx and reconstruct the path in Cartesian space with corresponding attributes.
2. Relax tangent vector on the path to simulate smooth camera behavior.
3. Calculate quaternion and up-vectors for correct camera rotation and translation.



(a) Detail: trajectory 1 in Houdini

(b) Detail: trajectory 2 in Houdini

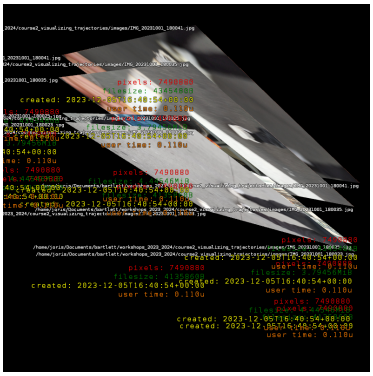


(c) Detail: smooth tangent vectors

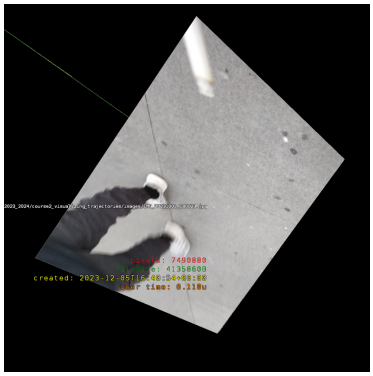
Figure 8: reconstructed .gpx data

#### 3.1.2.2 Reconstruct and Visualize Images from Metadata:

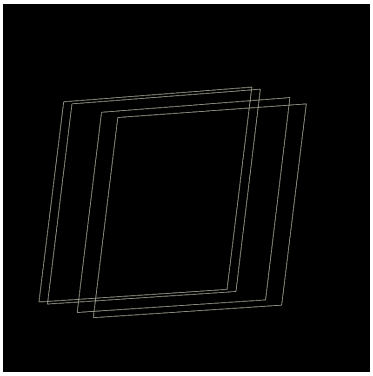
- Extract image metadata (e.g., resolution, user, date, latitude and longitude, altitude, time, pixels, etc.).
- Convert geographic values to Cartesian coordinates.
- Place, transform, and rotate images according to the camera direction.
- Apply the correct texture.



(a) Detail: images with attributes



(b) Detail: single image

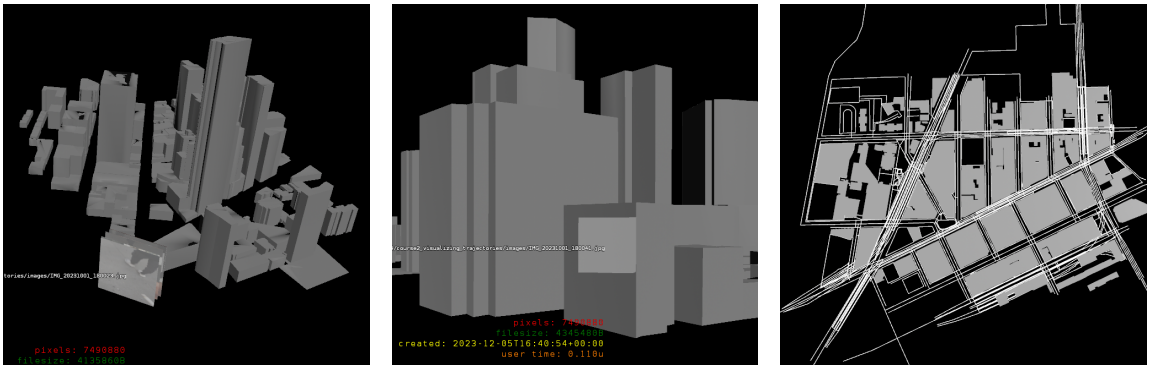


(c) Detail: wireframe

Figure 9: image previews

#### 3.1.2.3 Reconstruct .osm Data:

- Write a function that can convert latitude and longitude attributes to Cartesian positions.



(a) Detail: birds eye (b) Detail: perspective (c) Detail: road map

Figure 10: OSM extraction and positioning

#### 3.1.2.4 Combine all methods:

1. Make a new geometry container and reference the results of our manipulated OSM model, the reconstructed path, and the images.
2. Render out a frame sequence and compose it to an animation with ffmpeg.

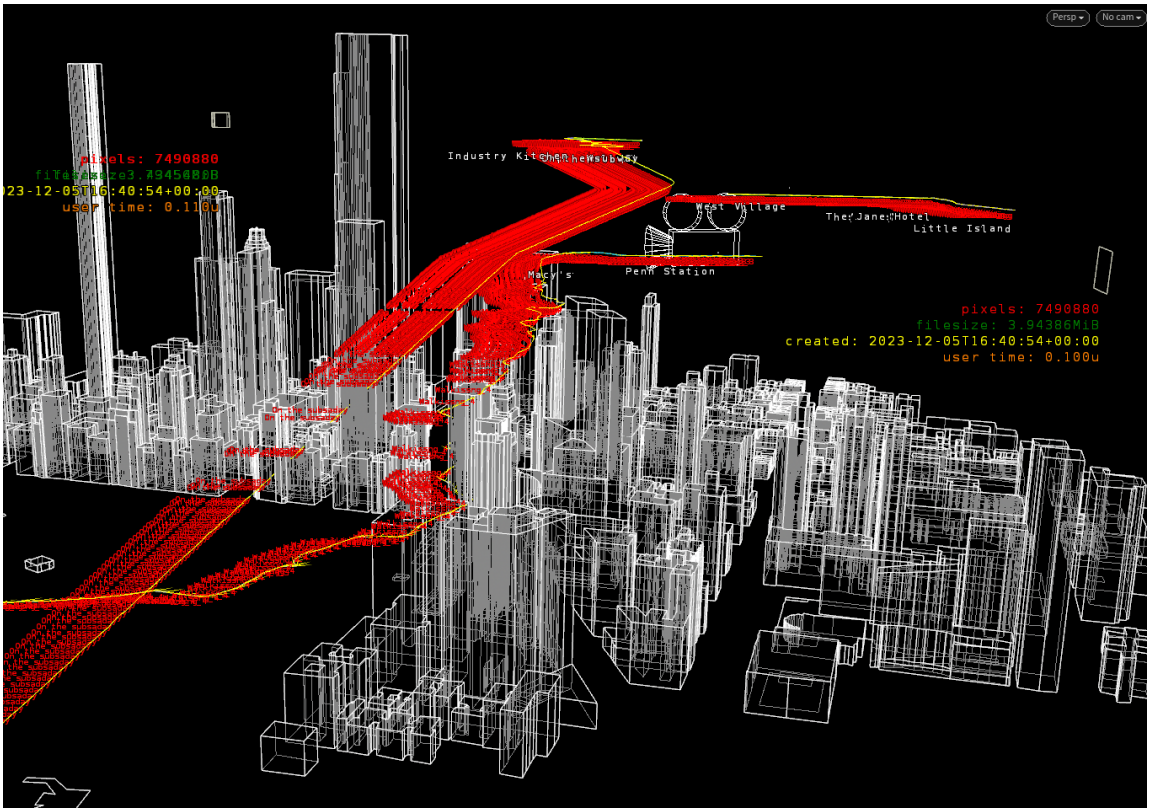



Figure 11: Visualization: Image, OSM, and GPX Data

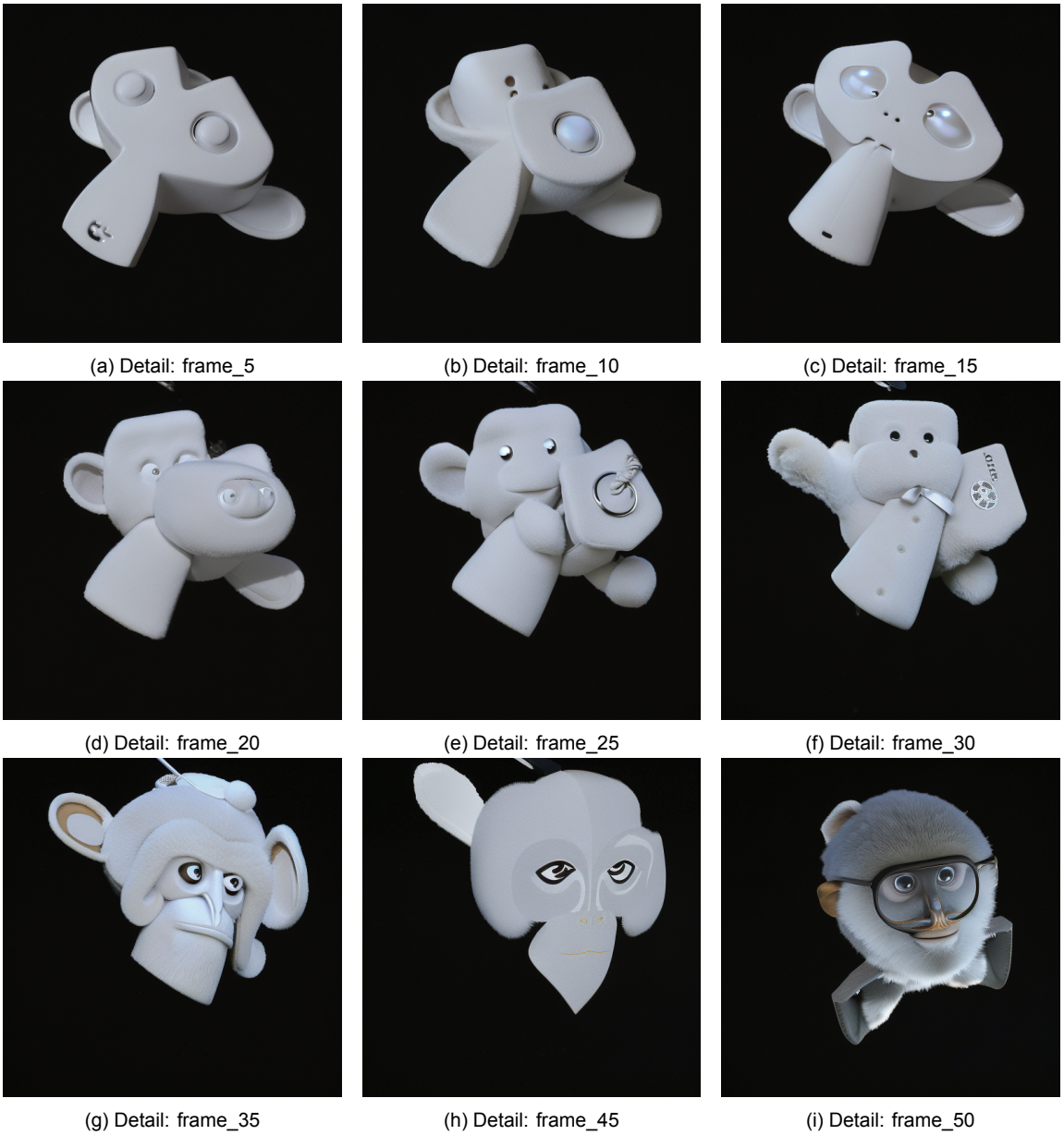
# 4 Workshop: Stable Diffusion in Blender

## 4.1 Overview

In this workshop, we are exploring stable diffusion models within Blender. It is a powerfull tool that can be used to quickly enhance your scene.

The example file can be downloaded from github.

1. Blender navigation and introduction.
2. Installing stable diffusion in Blender.
3. Prompting techniques.
4. Animating the prompts.
5. Rendering out the results and composing with FFmpeg.



(a) Detail: frame_5 (b) Detail: frame_10 (c) Detail: frame_15

(d) Detail: frame_20 (e) Detail: frame_25 (f) Detail: frame_30

(g) Detail: frame_35 (h) Detail: frame_45 (i) Detail: frame_50

Figure 12: stable diffusion results with animated weights. Prompts: monkey.