

RC11 Workshops

Joris Putteneers

houdini fundamentals

This session covers the houdini fundamentals. For the exercise, the following is expected.

- Alter 2 of the example setups we covered during class. This means, add / remove minimim 2 nodes to a setup. Changing the parameters is not enough. Adding a Null sop or adding unconnected nodes does not count.
- describe in the most detail what exactly your setup is doing. some important keywords and terminology are *node*, *parameter*, *procedural*, *datatype*, *attribute*, *class*, *geometry*, *point*, *primitive*, *vertex*,... Obviously, the example below cannot be used.

An example of such a description would be :

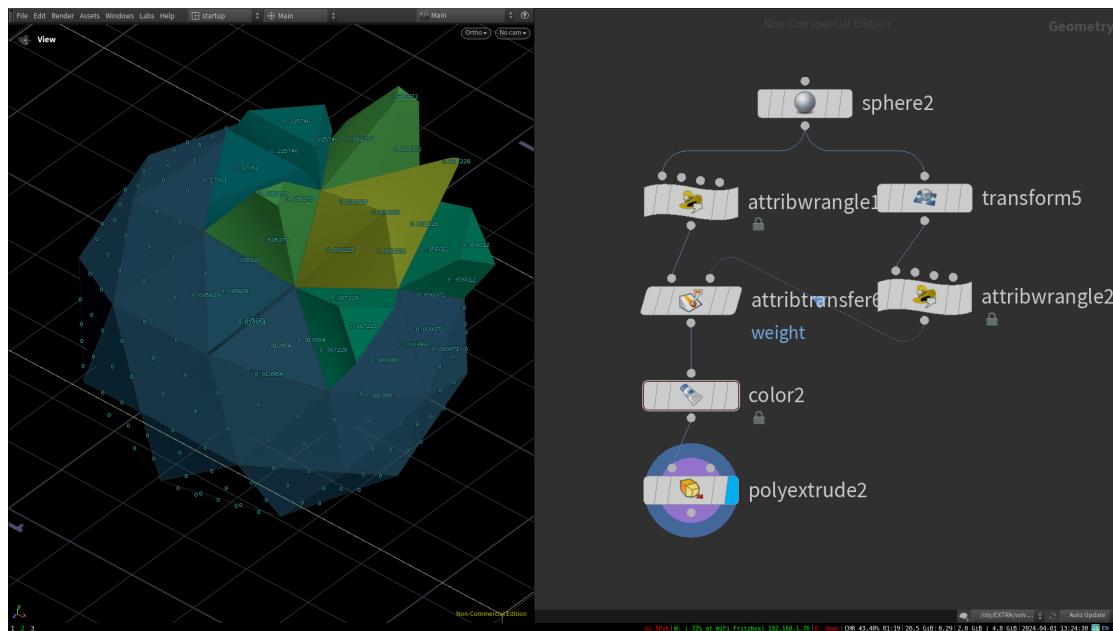
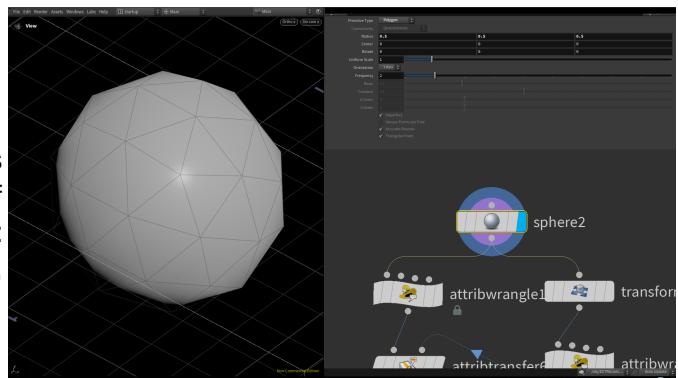


Figure 1: houdini fundamentals: setup 1 overview

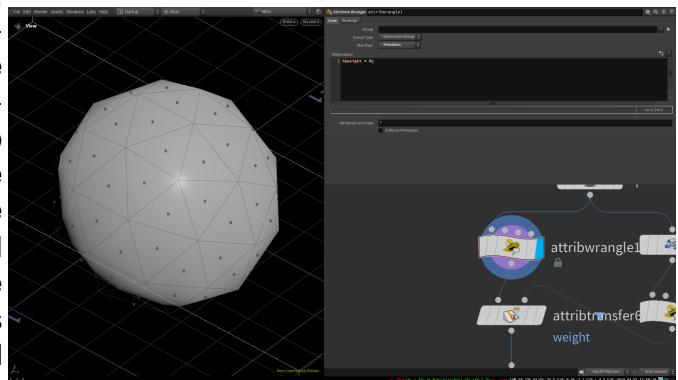
1: sphere SOP

A sphere is created. the sphere is of primitive type: polygon. Once of the basic datatypes that contains: points, primitives of type: polygon, and vertices.



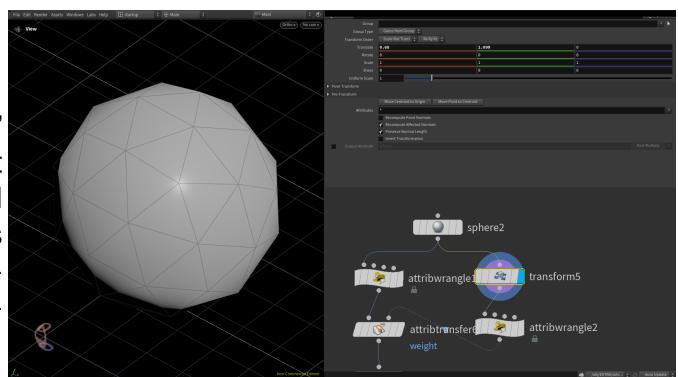
2: attribute wrangle SOP

We create a attribute wrangle sop, where we make a "weight" attribute. the attribute is of type float as described by the "f@" declaration. the attribute is set to run over primitives, this is because later in the node setup, we will use a polyextrude node where we will need a primitive attribute to drive the extrusion value. The value is set to 0. (which will be interpreted as 0.0 because of the float declaration) because later, we can interpolate between 0 and 1.



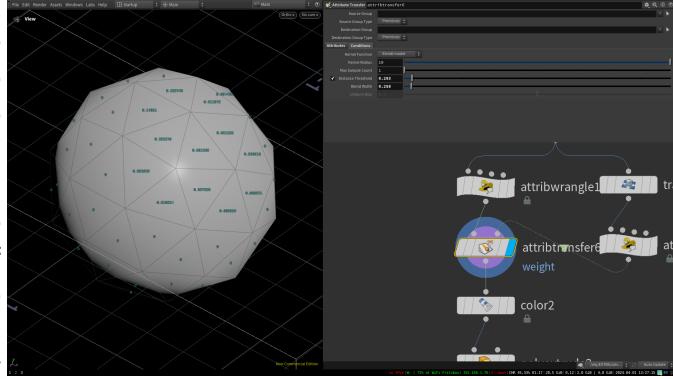
3: transform SOP

Then, i layed down a "transform" sop, This will be the attractor,it is just a duplicate of the original sphere. the distance between this duplicate (attractor) and the original sphere will dictate the final extrusion value.



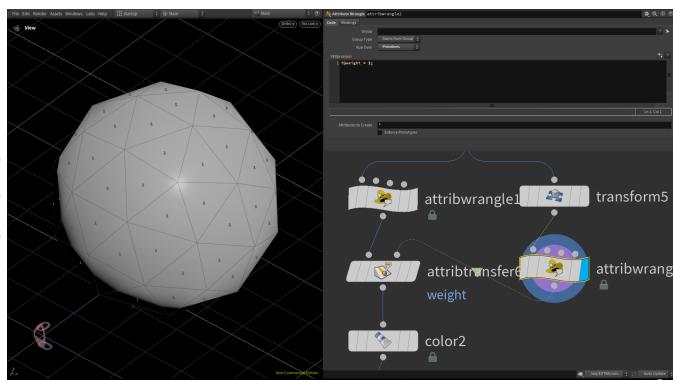
4: attribute transfer SOP

the attribute transfer sop uses the "weight" primitive attribute to write out the distance value between the original sphere and the attractor. by setting the originla 'weight' attributes as float, we can have value interpolations between 0 and 1. if we had set the weight attributes to integers, we could not interpolate. the output values would be binary (0 or 1).



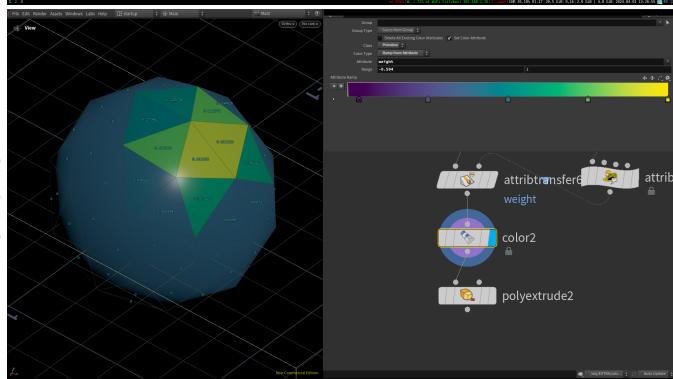
5: attribute wrangle SOP

the same as the previous wrangle SOP, only with the opposite value. The reason we interpolate between 0 and 1 is because it is easier to work with normalized values than arbitrary values.



6: polyextrude SOP

the same as the previous wrangle SOP, only with the opposite value. The reason we interpolate between 0 and 1 is because it is easier to work with normalized values than arbitrary values.

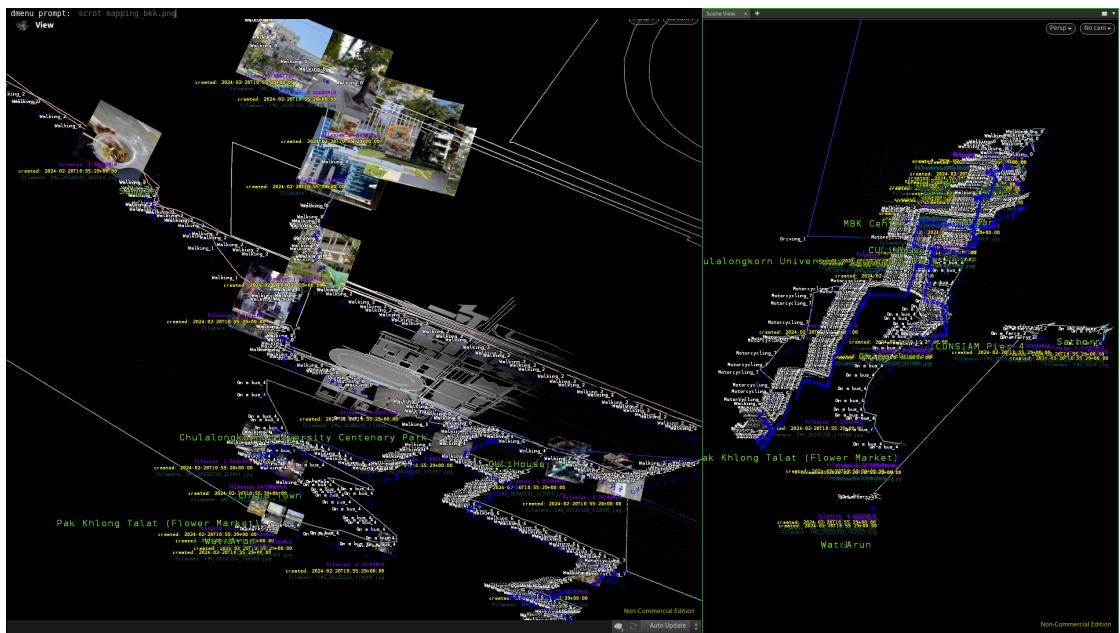


2. Visualizing gps and image metadata

This session covers how to overlay your google maps trajectories with images in houdini. For the assignment, the following is expected.

- load in your own images and gps data,
- generate a series of images depicting your travels.

An example of such a description would be :

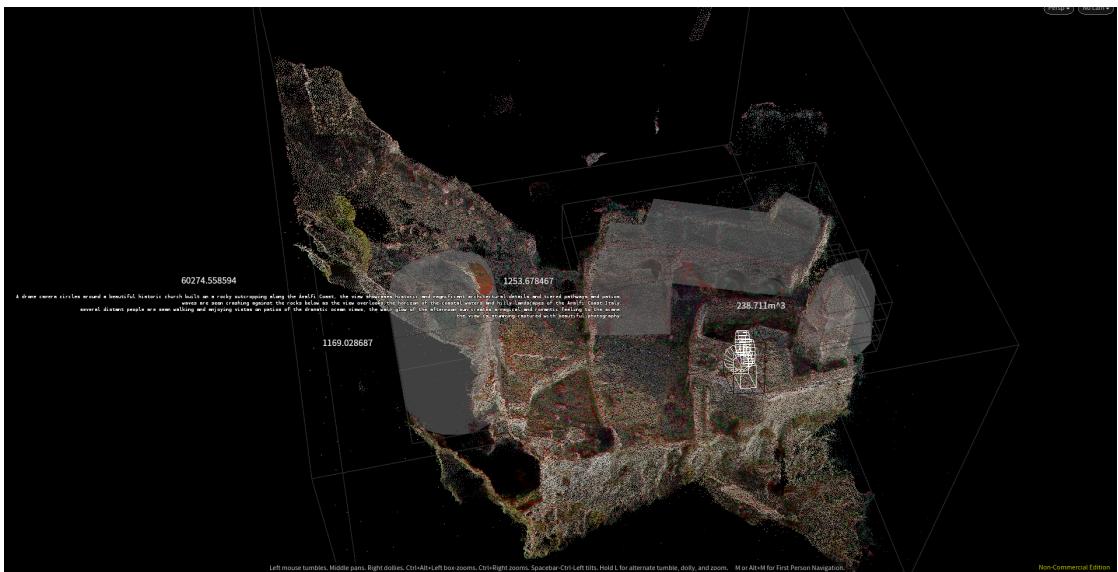




Sora to 3d model

In this workshop, we covered how to generate a 3d model from a video of a subject. In this case, this was the Sora AI video published from openAI. For the assignment, the following is expected.

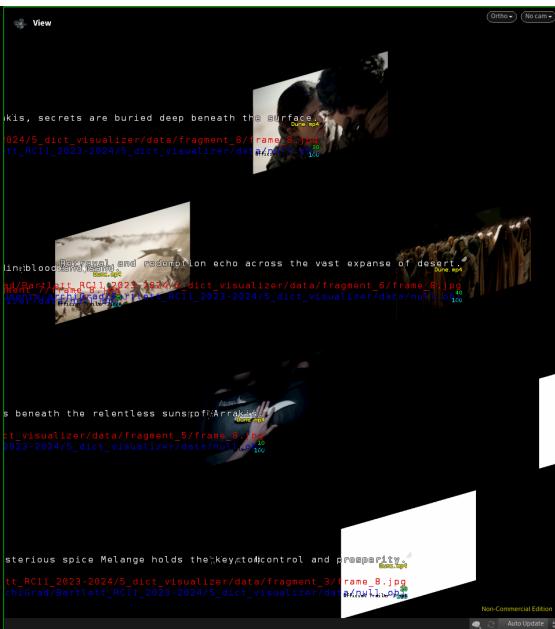
- Generate one or more photogrammetry model from a video/livestream/webscraped/google images/youtube. Take footage related to your projects.
- reconstruct your camera path.
- fond a way of adding information to the subject, an additional 3d model,) (volume specualtion, curavature, segenmbntation, adding you are sharing your entire screen sr, sharing —
- Visualize your model as a collection of images/renders.



visualizing json in houdini

In this workshop, we covered how to generate a json file that can be read and processed in houdini. This is usefull if you are constructing a .pickle or .json file and want to know how this can be visualized in Houdini

- based on your project/and other workshops, visualize a json file



The image shows a screenshot of the Houdini software interface. On the left, there is a code editor window displaying a JSON file with 33 lines of code. The JSON structure contains arrays and objects related to film clips and their frames. On the right, the main Houdini Viewport displays a 3D scene from the movie Dune. Several text annotations are overlaid on the 3D models, including "In the sands of Arrakis, the fate of ci", "A struggle for power and survival unfol", and "The mysterious spice Melange holds the". These annotations correspond to specific frames in the trailer, as indicated by the code in the editor.

```
1 [{}  
1   "folder": [  
2     {  
3       "paragraph": "isDistributed by\nNIPPON HERALD & OFFICE",  
4       "filmID": "Dune.mp4",  
5       "image_name": "frame",  
6       "model_path": "model.obj",  
7       "image_n": 25,  
8       "time": 100,  
9       "film_path": "Dune Part Two Official Trailer 3.mp4"  
10    },  
11    {  
12      "paragraph": "In the sands of Arrakis, the fate of ci",  
13      "filmID": "Dune.mp4",  
14      "image_name": "frame",  
15      "model_path": "",  
16      "image_n": 15,  
17      "time": 100,  
18      "film_path": "Dune Part Two Official Trailer 3.mp4"  
19    },  
20    {  
21      "paragraph": "A struggle for power and survival unfol",  
22      "filmID": "Dune.mp4",  
23      "image_name": "frame",  
24      "model_path": "",  
25      "image_n": 30,  
26      "time": 100,  
27      "film_path": "Dune Part Two Official Trailer 3.mp4"  
28    },  
29    {  
30      "paragraph": "The mysterious spice Melange holds the",  
31      "filmID": "Dune.mp4",  
32      "image_name": "frame",  
33      "model_path": "",  
34    }  
35  ]
```