

VINE GROWTH

Sanika Godbole | VSFx 350 | Project 3 | Spring 2025
05/29/2025 | Houdini Version: 20.5.278

Vine Growth User Guide & Breakdown



Render:

Renderer: Karma XPU

Average Render Time: 7 min/frame

Image Resolution: 1920x1080

Number of Lights: Skydome, 2 Sphere Lights, 1 Area Light

Sampling:

Primary Samples: 50

Min/Max Ray Samples: 1 - 9

Diffuse Quality: 10

SSS Quality: 8

Reflection Quality: 1

Refraction Quality: 1

Volume Quality: 1

Geometry Complexity:

Primitives: 997

Points: 997

ABSTRACT

The goal of this project was to procedurally animate growing vines. I combined my project VSFx 350 with my project from VSFx 425 so I could focus on look dev and composition as aspects just as much as the technical proceduralism aspects. My idea was to have the vines covering an abandoned building, with something someone has left on the windowsill once upon a time.

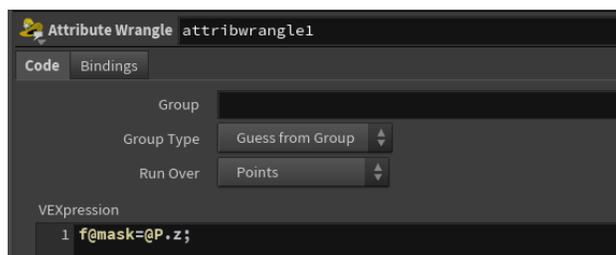


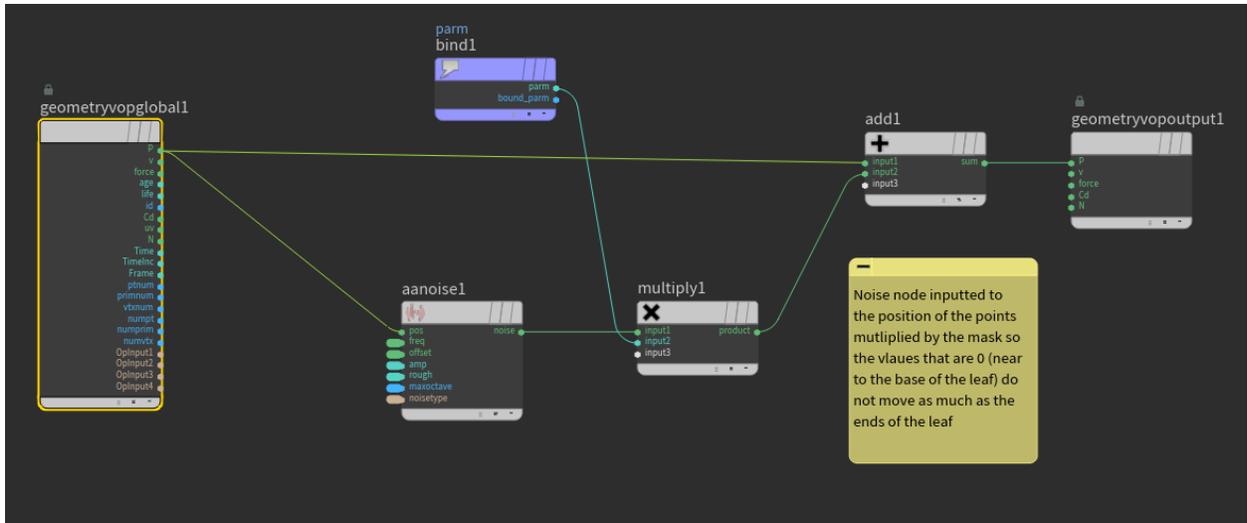
DETAILS

THE LEAF

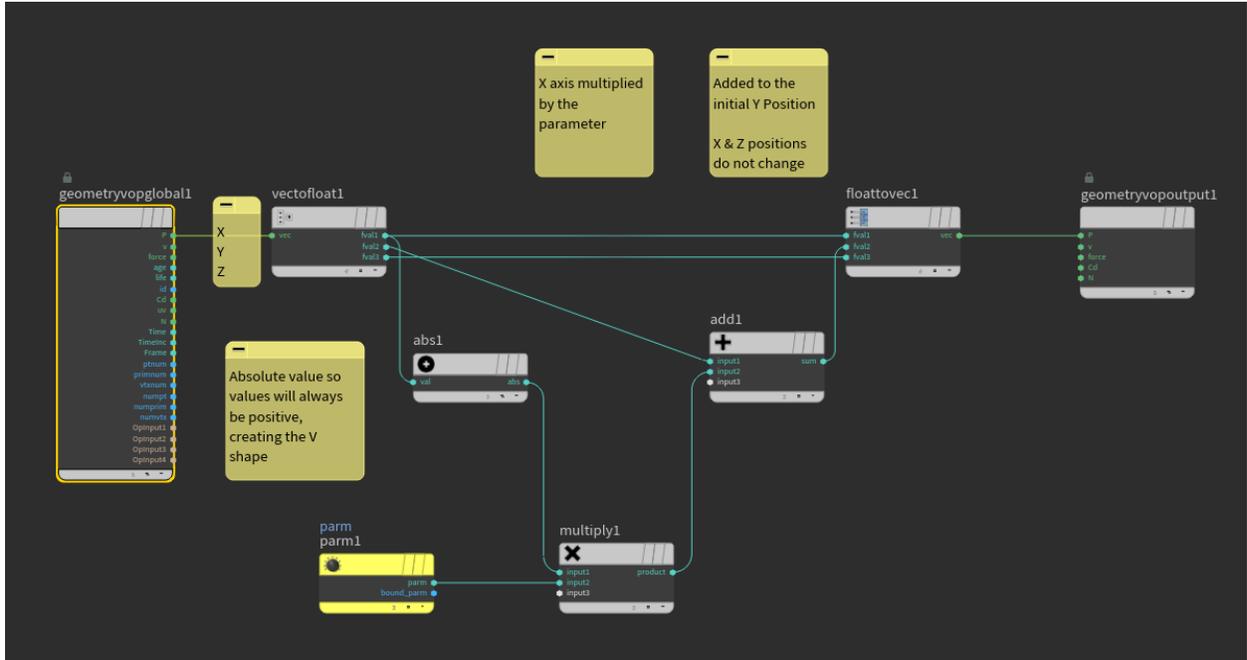
Here is the breakdown of my process. For guiding me through the entire process, I used this Youtube tutorial. <https://www.youtube.com/watch?v=GxKfKvYu7VU>. Since my knowledge of Houdini is really only of what I learned in this class, there were so many nodes and ways of approaching problems that I was only just beginning to realize and understand and following along with this video opened my mind to how one would approach certain obstacles.

The first step to creating my vine growth was to model the leaf. This part was very straightforward- I used a curve to draw one side of the leaf and then I mirrored it. I created a mask from the base of the leaf to the outside based on the position. This mask was used in a VOP network to add noise to the leaf in a way that it will keep the base of the leaf relatively stiffer than the ends of it.





Because each leaf is in a V shape as it grows open, I created another VOP network that would bend the other half of the leaf at the same value. As it would be on the negative side, I made sure that it was the absolute value of the original value.



Lastly, I did some more shaping tweaks. I keyframed the leaf growth which was just a scale from 1 to 10. While this is keyframed, it is later attached to the timing of the points on the stem moving, therefore it is still procedural because it is not binded to those specific keyframes.

THE STEM

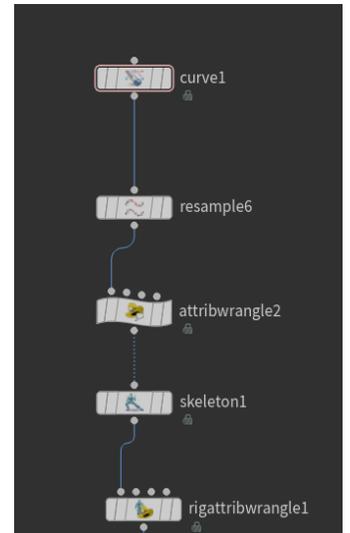
For the stem, I created a curve and resampled it which was used to create a rig skeleton and then a rig attribute wrangle. This is where I had to make my own tweaks to cater to what I want my vines to look like.

```

VEXpression
1 float angle = radians(chf("angle"));
2
3 float offset = lerp(0.2, -0.8, clamp(@Frame / 120.0, 0, 1));
4
5 float mask = f@curveu + offset;
6 mask = clamp(mask, 0, 1);
7
8 angle *= mask;
9
10 rotate(4@localtransform, angle, @V);
11

```

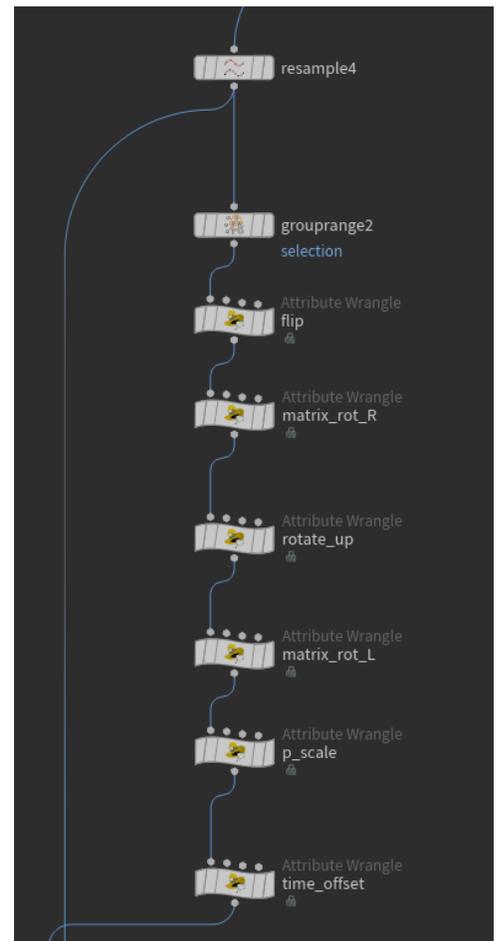
In the curve, I made sure to enable the curveuv attribute. The curveuv attribute ranges all the points in the curve within 0 to 1. When added to the offset the leaves slowly move from the position they are in over time. So technically, the leaves are sliding up the vine because their curve u position shifts over time. The lerp statement ensures that the offset shifts from 0.2 to 0.8 so no keyframes are needed. The last line is rotation on the matrix of the position of the point which means that instead of the geometry itself rotating, the point at which it is attached is rotation making the transition smoother.



In the next section, I resampled the curve again. This value is used to determine how many leaves I want on the vine. In the following attribute wrangles, the position of the normal of each vector is what is determined. I separated the points into two groups with alternating points. Then each group was assigned a range of angles that the normals would point to. This is what created the alternating leaves on the vine.

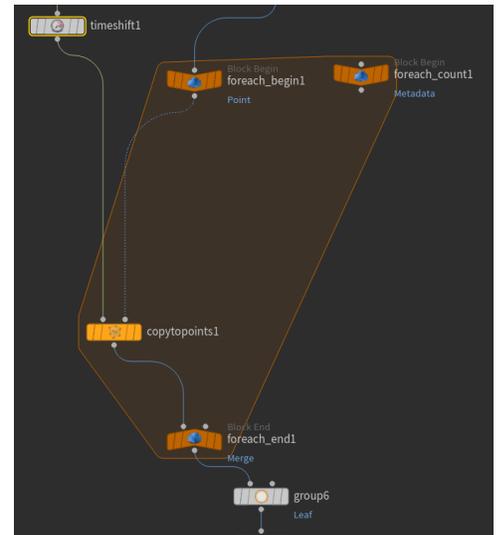
The time_offset attribute wrangle determines when the leaves start to unfold. Its metadata is used in a timeshift that is connected to an object merge that contains the leaf geometry. This is then connected to a for each loop with a copy to points inside. This is what allows a leaf to form at every point of the curve.

Because the copy to points is inside a for each loop, the rotation and size of every leaf is different which adds to the randomness of each leaf. The original curve that was



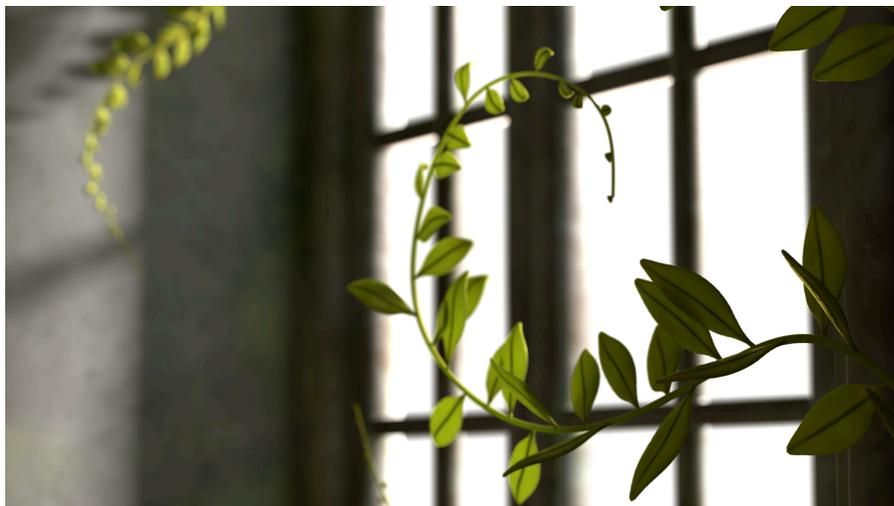
connected to create the skeleton was swept to create the stem of the vine. The leaves and the stem were then merged and ready to cache out.

I then copy and pasted this node network multiple times and changed the stem curve depending on where and how I wanted my vines positioned in the scene.



LOOK DEVELOPMENT

I knew I wanted my animation to look very cinematic, so I made sure to have high contrast to emphasize the focal point as well as depth of field. I added subsurface and a slight bump map to the leaves to make them glow in the sunlight. I comped in a sky in Nuke so that it would not add to my render time. I was playing around with different types of vines at first, but I ended up just sticking to the green leaf vines. In the future, I intend on making a texture map for every leaf. I have my primary light outside the window as well as a skydome, but I also have a sphere light inside the room so that I can control the contrast.



PROBLEMS & SOLUTIONS

One of the biggest issues I was having at first was that I was unable to write out a USD to put into the Renderfarm. The writing out was very slow and it would crash midway every single time. I later realized that I was not packing any of my geometry. I went back and within the copy to points made sure to pack and instance my geometry. I also made sure to pack my stem geo. This was also when I decided to cache out all my animation, so that it would theoretically make my file and render faster. Now I was able to write out a USD file, which led me to find another issue. My animation did not look correct when I brought it back into the file as a USD. I tried different ways to import my caches into Solaris including file reference, file asset reference, sop import, geo cache node, and each way had different problems. For some of them, the vines grew backwards and for others the stem grew without the leaves. In the end what worked was just a regular sop import which was pointing right to the OUT null, loaded as reference, and with animated topology attributes turned on. This solution was found as a result of lots of trial and error and constant rendering with the command line. This was what made me realize how powerful the command line was because I could test my render without putting it through the farm.

Another issue that I had was that after I exported my uvs, painted a texture map, and plugged it in, it looked as if I had never uv'd in the first place. I still do not know why this was happening. My guess is that perhaps the Auto UV changes the placement of the uvs everytime the file is opened. At this point of time, I did not want to spend time repainting my texture so my solution was to export the mesh that I painted on from Substance Painter and do a file import back into my node network.

A non technical issue that I had was that for the longest time I could not get my composition figured out. I played around with adding different props, widening the space, closing in on the space. The main issue was that I could not find a way to fill up the walls on the sides of the window and the scene looked very empty. I decided to create two vines right in front of the camera with depth of field so the viewer will not be able to tell what is happening behind them while also framing the focus point of the scene.

While making the vines, my vines were clipping through the walls as they grew. They were also not growing in the same path as what my curve was. While looking through all my attribute wrangles, the reason this was happening was that it was growing in the direction of its normals at that time. I deleted that variable in my code, and it started growing in the direction of the path of the curve, making it predictable where my vines would clip through the walls so that I could adjust the curve.

A problem that I have not found the solution to is adding a gobo to create god rays. I got the general atmospheric working with a lightfilterlibrary and a karma fogbox, however, I am still unable to get the rays that would result from a gobo.

One last problem that I have still not found the solution to is that even after I tested to make sure my render worked in husk mplay and usdview, after it goes through the farm, the stems to the vines do not show up. I sent a problem report about this issue and although they claimed to have fixed it, it does not seem to be working on my end. I am still currently going back and forth to fix it. Although I probably will not be able to do a resubmission before the deadline, I do intend to polish this project over the summer so that it will be a strong piece in my demo reel.