

Singing Dogs

A Mix of Random Shader Effects

Sean Song
CS: Game Design
University of California,
Santa Cruz

Sandra Tiu-Latt
CS: Game Design
University of California,
Santa Cruz

Lily Young
Art & Design: Games and
Playable Media
University of California,
Santa Cruz

ABSTRACT

In our project, we wanted to do something simple, fun, but allowed us to highlight several different components.

The creative vision for the project was to have two dogs barking at each other based on the music that was playing. As the dogs bark, they would emit smaller dogs and sparkles coming from its mouth. The goal that we were going for was a light-hearted, cute scene with dogs.

Our basic components include an audio listener and a particle system. But we *jazzed* it up with several shaders in order to create fun visual effects.

1 Creating the Models

As a 3D modeler, my technical challenges were making the model perform well. This shiba model was particularly challenging for me because it was my first organic model, so I made the model not function how it is supposed to. The problem was that the faces and vertices of the model were acting strange like some were appearing inside the model and it was connecting in a strange way. Some of my colleagues had to help me fix it, so it became properly made. The solution to this problem was to cut the model to delete all the extras inside and remirror it. I feel like this problem happened because I mirrored it wrong the first time. When I made the sparkle, it caught me off guard. I knew what a sparkle looked like in 2D but somehow I could not visualize making a 3D one. When my colleagues explained to me, I began to understand. They wanted a sparkle that was equal on all sides whether it be on the

X-axis, Y-axis, or Z-axis. When I began implementing it in Blender, it was challenging because it did not work if I just simply mirrored it. The solution I found to this problem was to delete all the faces and keep one outline of the shape and copy and paste it manually.

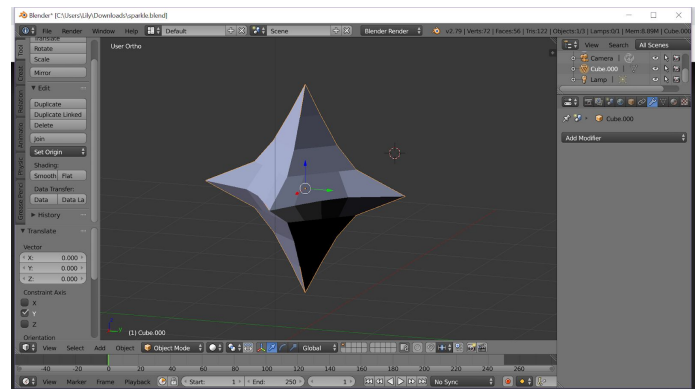
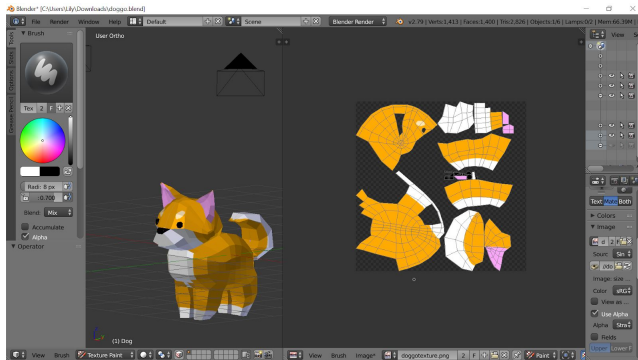


Figure 1: Sparkle Model

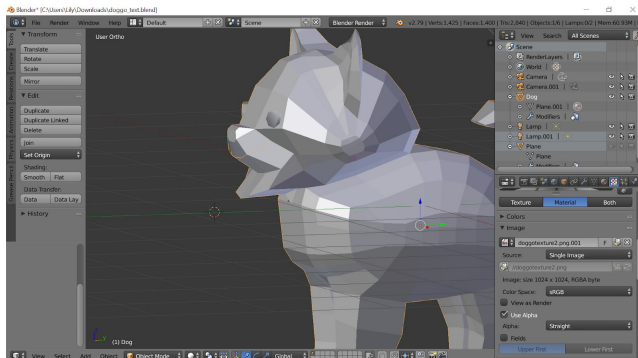
1.1 The Shiba

In order to use the model in Unity successfully, we had to make sure the textures stayed properly. Originally, we had colored the dog with materials in Blender. However, we realized that was a mistake in Unity as it did not stay. Instead we chose to UV unwrap the dog and create a texture so that we could just import the texture along with the model in to Unity.



Originally we wanted it so that the dogs would have barking animations in tune to the music, however, as we tried to animate—nothing would work.

We tried rigging but none of the bones would stick to the mesh, and without it, you couldn't animate just the mouth. However, we decided that we would be fine without the animation as the focus was on the other effects in the scene.



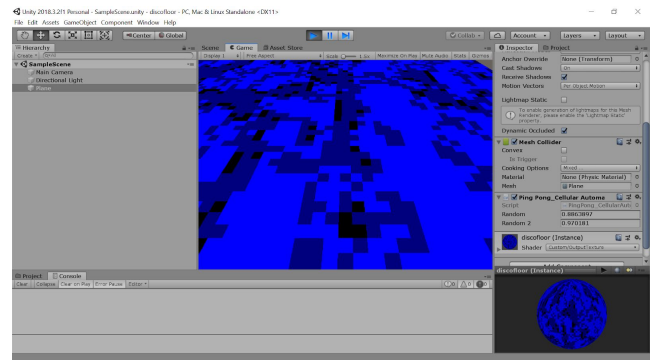
2 Setting the Scene

In order to achieve our goal, we needed to set the scene. Since we wanted the dogs to bark out to music- we also wanted to include a disco floor and ball.

2.1 Disco Floor

Originally we planned to create a cool, eclectic “disco” floor using the Game of Life algorithm we learned this past quarter. We modified it so that the colors were different, and tweaked numbers so that there were in

between shades of black and white. Ultimately, we decided this disco floor was too busy and scrapped it.



2.1 Disco Ball

The disco ball uses a rainbow shader that interpolates between varying colours using a sliding H value in an HSL colour calculation. It also allows for variable settings of intensity (as shown with the pastels of the disco ball), luminosity (with the emissive glow of the sparkles), and speed of colour transition (with the disco ball's high speed).

3 Adding the Effects

Now, on to the fun stuff! We added several different components in order to make the singing dogs stand out. We added a shader to make the sparkles rainbow, we added a bloom filter to make them glow, and we added a toon shader to create a cartoon effect.

3.1 Toon Shader

The toon shader degrades the default lighting model into larger, single step jumps that define a clear change in shading along different normals. In addition, the anti-aliasing of the cel shading also removes a jagged edge left from pixel-smoothed steps that would be present on rounder models but as ours are low-poly and have distinct changes in surface normals, such a border is not present.

3.2 Rainbow Shader

Similar to the shader used for the disco ball, the sub-emissive sparkly models of the trails from the smaller dog particles also linearly interpolate hues along an HSL spectrum. The key difference here is that the sparkle particles have high emission values that are attuned to the bloom filter that is applied to the camera, allowing them to appear glowing and shinier than the disco ball and the rest of the scene.

3.3 Bloom Shader

The camera-field applied bloom filter blurs any bright emission objects above a certain threshold, downsampling the quality until reaching a desired intensity and then re-upsampling those layers back onto the original view, resulting in a combined colour that appears greater than what the computer can actually display. In addition, it also features a soft knee curve threshold that radiates the blurred glow and softens the transition between the glowing objects and their surroundings.

4 Putting it All Together With Audio

The audio component of the scene was created using a basic FFT analysis of our chosen audio, causing the particle systems to emit whenever the volume exceeded a certain threshold. The selected piece was a cover of Queen's *Don't Stop Me Now* sung by an internet celebrity dog, Gabe the Dog. The controlling script also has a buffer timer between emissions in order to limit the amount of active particles and alternate between the two opposing dogs and their subsequent emitting particle systems.

REFERENCES

[1] FNM Surname (2018). Article Title. Journal Title, 10(3), 1–10.
[2] F.N.M Surname, Article Title, <https://www.acm.org/publications/proceedings-template>.
[3] F.N.M Surname and F.N.M Surname, 2018 Article Title, The title of book two (2nd. ed.). Publisher Name, City, State, Country.
[4] Ian Editor (Ed.). 2018. *The title of book two* (2nd. ed.). University of XXX Press, City, Chapter 100. DOI: <http://dx.doi.org/10.1000/0-000-00000-0>.

