3D 4D Ray Tracer

Sergey Barabanoff
CSCI 5607

Project Overview

Ray tracers are used to project 3-dimensional objects onto a 2-dimensional screen in such a way that depth, lighting, and object appearances are preserved, causing the image to feel almost tangible. My goal is to explore whether a ray tracer can be used to visualize higher dimensions. Specifically, my project explores 4-dimensional objects like hyperspheres and hyperplanes. Of course, a ray tracer cannot provide a perfect visualization of such objects, because it is impossible to see in more than three dimensions at once, but a ray tracer should be able to visualize some of the oddities that arise from objects and light existing in an additional dimension.

However, without any sort of movement or repositioning, 4-dimensional visualizations appear as static 3D scenes. To account for this, my project also includes a red-blue anaglyph effect to create the feeling of three dimensions extending out of the two dimensional screen. This will provide additional depth to scenes that might help the fourth dimension become more tangible, and also help understand how 4D objects change as their three dimensional projection changes.

While the project has come to fruition, I have found it difficult to create scenes that appear 3D but also feel as though there is a fourth dimension rendering. There were a handful of other obstacles that made this implementation difficult, too. First of all, the previously provided PGA library for project 3 uses multivectors that only work for three dimensions. Rewriting this would require a significant effort, so I opted to use the Gaalet library, which has 3D PGA that is easier to extend to a fourth dimension. Also, I couldn't quite figure out how to project 4D objects into three dimensions in an interesting fashion. Since I plan to add stereoscopic rendering, I opted to have my projections be based on the camera's position in the fourth dimension. Objects are scaled based on their distance from the camera (similar to how farther objects appear smaller). Then, stereoscopic rendering is created by adjusting the position of the camera in the fourth dimension, altering the 3D projection of these objects.

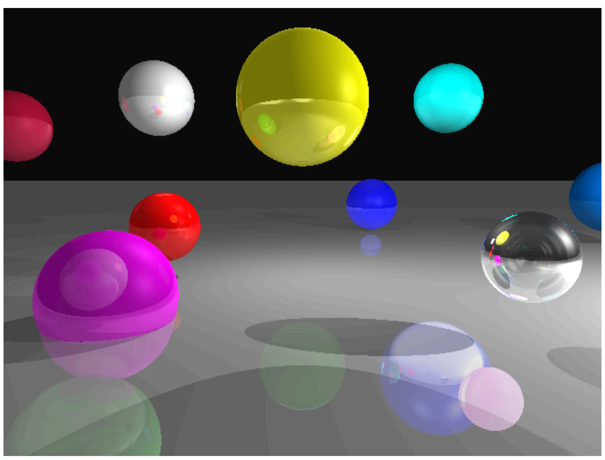
The last obstacle I've encountered is that I have found it very difficult to create scenes in 4D that look interesting upon ray tracing. Sometimes objects just aren't projected into 3D, so it has been difficult to come up with good scenes.

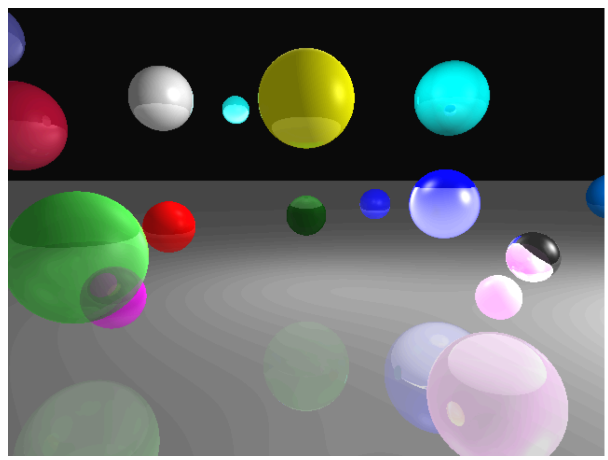
Connection to Our Class

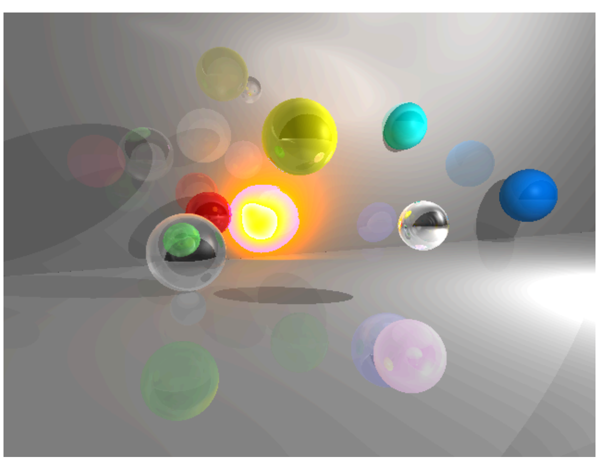
My project is heavily inspired by project 3. This project incorporates aspects of topics like the rendering pipeline and lighting models, but the ray tracer also incorporates more conceptual topics like PGA. Also, I implemented one of our more recent topics, which is stereoscopic rendering for my 3D effect. When we discussed this in class, we described how red-cyan 3D glasses are used to provide each eye with a different image, creating a visual offset with a feeling of parallax. We also used 3D glasses so that we could actually experience this, and if you still have them, I recommend you use them to view the images later in this report.

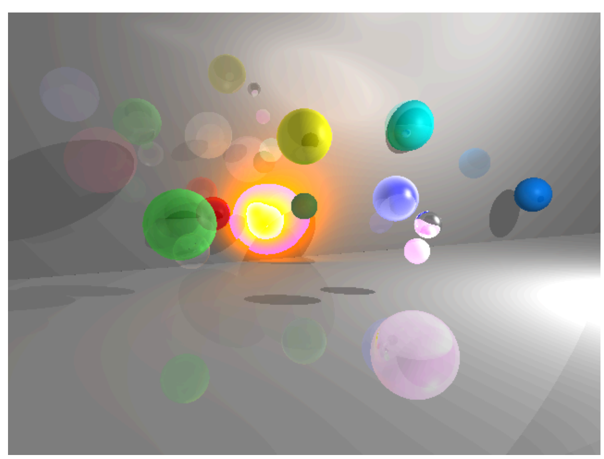
Key Features

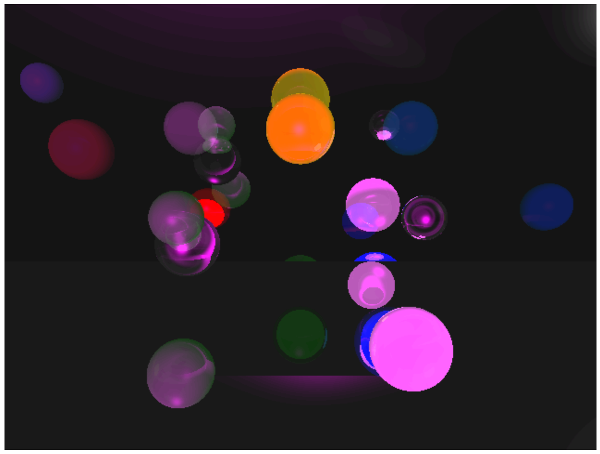
My project allows users to create a 4 dimensional scene with hyperspheres (four-dimensional spheres) and hyperplanes (four-dimensional planes). This ray tracer also allows for ambient lighting, point lights, directional lights, spotlights, and different materials. The Gaalet library is used to implement four-dimensional projective geometric algebra for objects and lighting (although light sources are constrained to a 3D space due to the lighting model). Ray tracing functions by ray tracing 4D rays and projecting hitpoints into 3D, while supporting shadows, reflections, refractions, and specularity. The ray tracer uses parallelization via OpenMP and supports a 4D camera position via a camera_w parameter, which is what is adjusted when rendering with an anaglyph effect for a 3D effect.







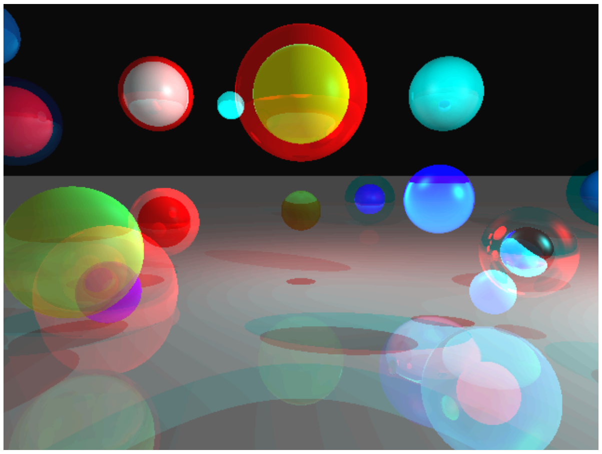
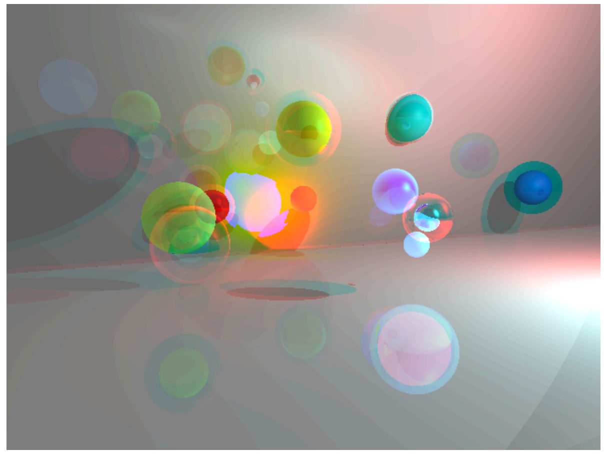
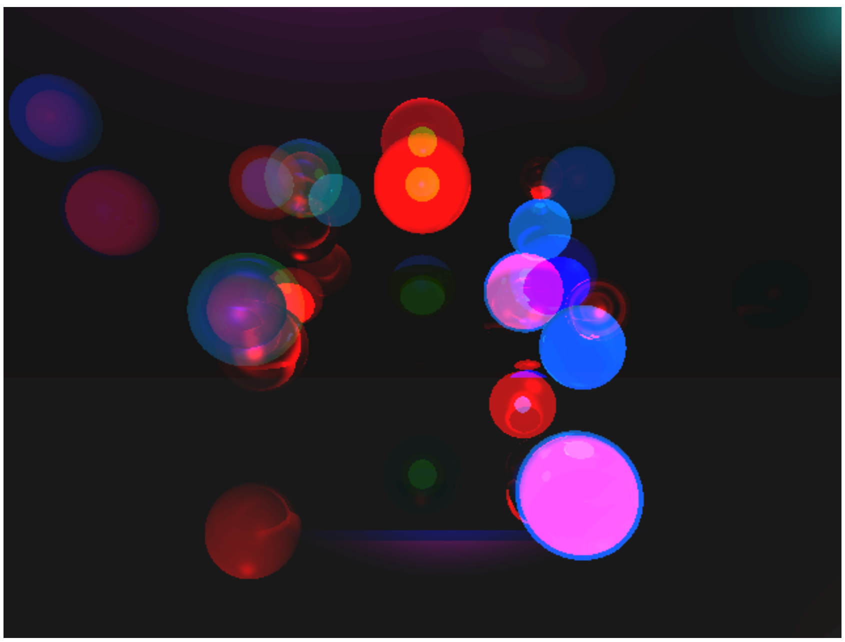




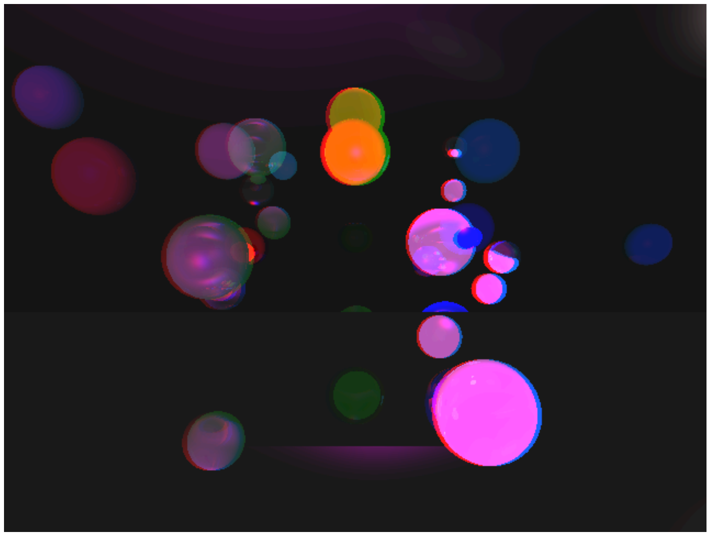
The images above are all from my 4D ray tracer, without the anaglyph effect. As expected, they are somewhat unremarkable when considered in isolation. What is quite interesting, however, is how they change as the camera position is adjusted in the fourth dimension. Objects change sizes, appear, and disappear based on the camera's position. What is also very interesting are the visuals obtained from reflections and refractions. Sometimes, objects are visible only in reflections and not in the actual scene, demonstrating that they are there but not in the 3D projection. This can be seen in the two images from the first scene, which differ by only 0.5 in the fourth dimension. The first image shows the green sphere reflected on the purple sphere, but it isn't visible as an actual object in the scene. In the next image, the green sphere appears and the purple shrinks. The gray sphere on the right also has some very interesting ripple refractions in the first image, but then shrinks and refracts almost only light from the purple sphere.

The second scene is another example of how perspective causes hypersphere projections to change significantly. In the left image, there is a sphere reflecting a green sphere in clear detail, but the green sphere is nowhere to be seen in the image as a visible object. In the right image, the green sphere reveals itself while the original sphere is no longer present. There are similar scenarios throughout the scene. Again, an interesting phenomenon occurs with the lighting. This scene is supposed to have a spotlight shining on the plane in the back. However, this spotlight did not appear as expected when repositioned. Instead, it looks as if it is also undergoing interesting refractions inside of the overall hyperplane that creates the back plane, not just its projection. Although it might look like it, there is not a sphere behind the plane where the spotlight is shining; the rings that form at the spotlight (yellow interior, bright white ring, lighter outer rings) seem to all be occurring because of unusual refractions, as far as I can tell, and not from any spheres in the scene.

The image at the bottom is the most remarkable, in my opinion. This image uses a set of spheres placed in front of two hyperplanes, one that is orthogonal to the y-axis and one that is orthogonal to the z-axis. They have a mirror-like texturing that reflects objects very clearly. The scene also has a directional light pointed forward to the right. With close inspection, it is clear that several objects appear only in reflections. Also, the internal refractions seem to cause interesting effects. The bright pink sphere in the front of the scene is not supposed to be glowing pink; what I suspect is happening is that the refractions inside the hypersphere from the directional light are creating a caustic effect because the light is refracting in complicated ways. Another sphere on the right undergoes that same ripply effect as before and some spheres look like they are focusing the directional light into a single point.

Here are those same scenes but now using stereoscopic rendering to visualize them in 3D. While I believe they look interesting, I've come to think that this sort of 3D effect is ill-suited for this context because the projections vary significantly between views. Instead of creating a sense of parallax, the 3D effect with the glasses creates a lot of binocular rivalry, so only one perspective is visible at a time.



For reference, here is one image with an offset along the x-axis rather than the w-axis. There is much less binocular rivalry here because there aren't any distortions.

Resources Used

The majority of my code was based off of my project 3 code. As mentioned, I utilized the Gaalet library to implement 4D PGA and I used OpenMP for parallelization. Regarding my use of AI, I used AI to help me refactor the PGA library from project 3 to be compatible with Gaalet and 4D PGA. This was necessary because I have not previously used Gaalet and I feared that developing my own 4D PGA with that library would cost me too much time. The Gaalet library can be found at this repository: https://github.com/icaven/gaalet

Future Work

The most interesting extension for this project would be to use an animated rendering where the user can move through the scene. This would provide a much more concrete visualization of how 4D objects change as the camera position changes. In combination with the anaglyph effect, this could be a very interesting visualization of 4D objects. This might also be better suited for stereoscopic rendering because the perspective offset could be in the 3D space, preventing any distortions between the two perspectives. Similarly, it would be interesting to introduce different kinds of 4D objects aside from hyperplanes and hyperspheres. Perhaps some sort of 4D tetrahedron could be used to make up for the 3D triangles that were not included in this project; other objects like tesseracts (4D cubes) could also be implemented.