

Documentatie

ChristmasSpirit

Proiect realizat de
Ilie Dumitru
Neculae Andrei-Fabian

Proiectul este o simpla scena de Craciun prin care camera se misca pe o trajectorie predefinita sau, utilizand tasta 3, poate intra in modul miscare libera.

Scena este formata din mai multe obiecte importate din fisiere .obj, incarcate in mod asincron, create cu ajutorul aplicatiei Blender sau descarcate de pe internet si texturate folosind Blender. Scena contine de asemenea 3 ferestre in spatele carora am simulat un efect de ninsoare folosind tehnica de randare instantiata si tehnica utilizata in proiectul 2D pentru desenarea unui cerc folosind un triunghi. Camera se misca pe o trajectorie formata din mai multe curbe Bézier.

Proiectul este original deoarece combina mai multe idei pentru crearea unei scene reusite. Mai exact traectoria formata din curbe Bézier, efectul de ninsoare si modul de miscare libera.

Capturi de ecran din aplicatie:

Cod randare scena:

```
void DrawScene()
{
    Utils::UpdateCamera(
        pathMap["camera"].interpolate(Utils::sceneTime),
        glm::normalize(pathMap["camera_orient"].interpolate(Utils::sceneTime))
    );
    Snow::Draw();
    Snow::UpdateTranslations();
    scene.draw(meshMap);
    Utils::UpdatePathTime();
}
```

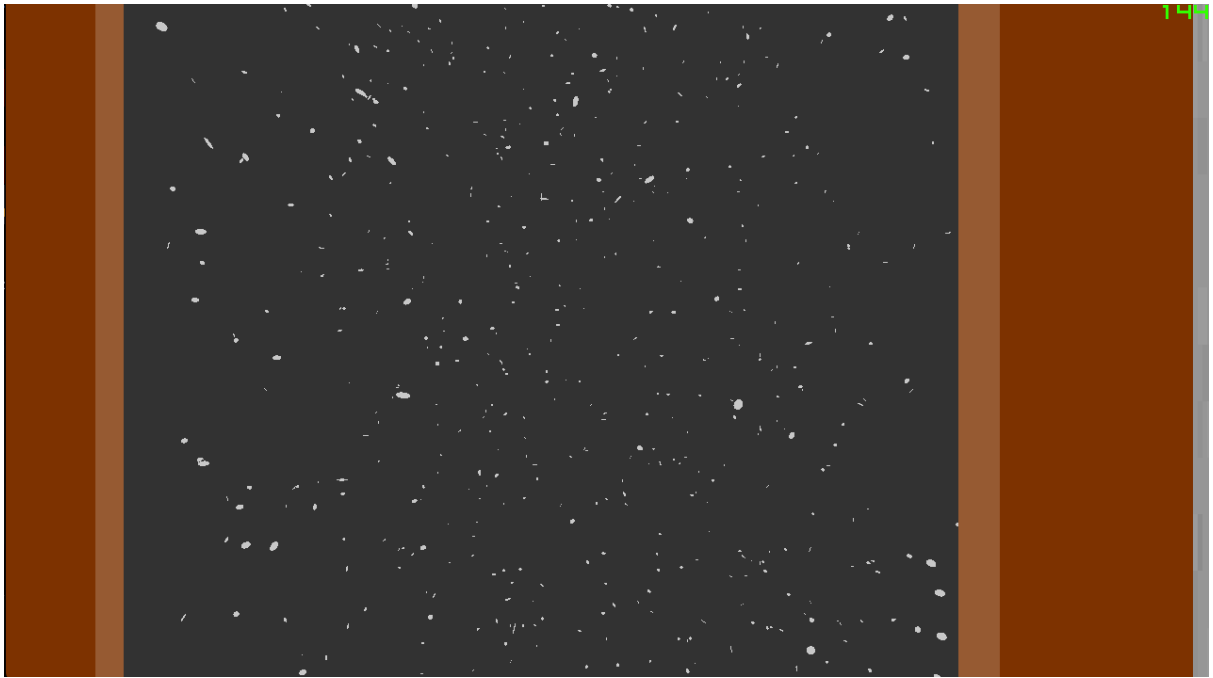
Cod incarcare asincrona:

```
std::mutex pathMapMutex;
std::mutex meshMapMutex;
try
{
    for (const auto& path : fs::directory_iterator(BEZIER_PATH))
    {
        assert(path.is_regular_file());
        std::string name{ Utils::DropFileExtension(path.path().filename().string()) };
        futures.push_back(std::async(std::launch::async, [name, &pathMapMutex] {
            Path data{ Path::readFromFile(Utils::FILE_PATH("path", name)) };
            std::lock_guard<std::mutex> lock(pathMapMutex);
            pathMap[name] = std::move(data);
        }));
    }
    for (const auto& mesh : fs::directory_iterator(MESHES_PATH)) {
        assert(mesh.is_regular_file());
        std::string name{ Utils::DropFileExtension(mesh.path().filename().string()) };
        futures.push_back(std::async(std::launch::async, [name, &meshMapMutex] {
            Mesh meshData;
            meshData.loadMesh(Utils::FILE_PATH("mesh", name));
            std::lock_guard<std::mutex> lock(meshMapMutex);
            meshMap[name] = std::move(meshData);
        }));
    }
    for (auto& future : futures)
    {
        future.get();
    }
}
```

Scena in aplicatie (la un moment dat):



Efectul de ninsoare:



Scena in Blender:



Contributii individuale:

Dumitru a creat traiectoria camerei, incarea obiectelor mai complexe (care nu sunt formate doar din triunghiuri), cateva modele si scena facute in Blender si incarcarea scenei dintr-un format intermediar in aplicatie. Andrei a realizat efectul de ninsoare, incarcarea asincrona a obiectelor, arhitectura proiectului si obtinerea modelelor pentru Mos Craciun, copil, cadouri si brad.

Resurse:

1. [Copil](#)
2. [Mos Craciun](#)
3. [Brad \(fara globuri\)](#)
4. [Cadou 0](#)
5. [Cadou 1](#)
6. [Cadou 2](#)
7. [Cadou 3](#)
8. [Cerc din triunghi](#)

Anexe:

[Video demo](#)

[Link la repo](#)