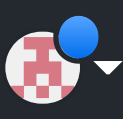




Search or jump to...



[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)



CristianQiu / **Unity-Flocking-CPU-GPU** Public

Watch

3



Fork

3



Star

21



[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)



master



1 branch



0 tags

Go to file

Add file



Code



About

CPU and GPU flocking implementations in the Unity game engine. Based on Unity's ECS implementation using DOTS, presented by Mike Acton.

unity

multithreading

flocking

compute-shaders

unity-ecs

Readme

View license

21 stars

3 watching

3 forks

Releases

No releases published

Packages

No packages published

Languages



CristianQiu License, again...

2f97cef on Apr 23, 2021

60 commits



Assets

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



GithubImgs

Removed image

15 months ago



Logs

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



Packages

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



ProjectSettings

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



UserSettings

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



.gitignore

Updated gitignore to exclude vscode stuff. Deleted Rider stuff

3 years ago



.vsconfig

Updated the almost two year old implementation of ECS from Unity's us...

15 months ago



LICENSE.md

License, again...

14 months ago



README.md

Update license, readme. Get rid of redundant folders, Updated engine ...

14 months ago



README.md

About

The project contains different CPU and GPU flocking implementations in Unity's game engine, using a single threaded and multithreaded approach in the CPU, as well as another using compute shaders. All of them are mainly based on Unity's ECS implementation (which was presented by Mike Acton in GDC) and is also included in the project. The original Unity repository that includes the boids implementation using ECS can be found at <https://github.com/Unity-Technologies/EntityComponentSystemSamples>. All of the art assets are from Unity's demo. See Mike's technical presentation at <https://youtu.be/p65Yt20pw0g>.

I started this project in 2019 when Unity had already been advertising for about a year their new tech stack: DOTS. I wanted to have an overview on what data-oriented design along with the job system and the burst compiler was able to do in terms of performance, so I decided to compare Unity's implementation against other paradigms, replicating their algorithm as much as I could.

NOTE: As of 22/04/2021 I have upgraded Unity's demo to its latest version, since the project was stuck to Unity 2019.1.14f1 and an older version of the DOTS implementation. I recommend using Unity 2020.3 LTS since future versions may break the project. I believe the core of the algorithm has not changed and most of the new stuff is syntactic sugar and new tools.

Preview

The following gif showcases over 500k boids being simulated in compute shaders with a 980ti at 1080p in a build.



© 2022 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Docs](#)

[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

[Blog](#)

[About](#)