

2D Stable Fluid Simulation

106062120 史孟玄

106062216 馮謙

Terms in Implementation

1. Advection

計算 fluid 位移的步驟，將每個 pixel 視為獨立個體，計算其下一個時間點的位置，由於 forward tracing 容易導致有些 pixel 空洞或是重疊，因此用 backward tracing。

2. Pressure and Diffusion

計算液體受壓力自然流動，為了平行畫運算，採用 jacobi 演算法來解出每個點的壓力。

3. External Forces

外部施加給 fluid 的力。

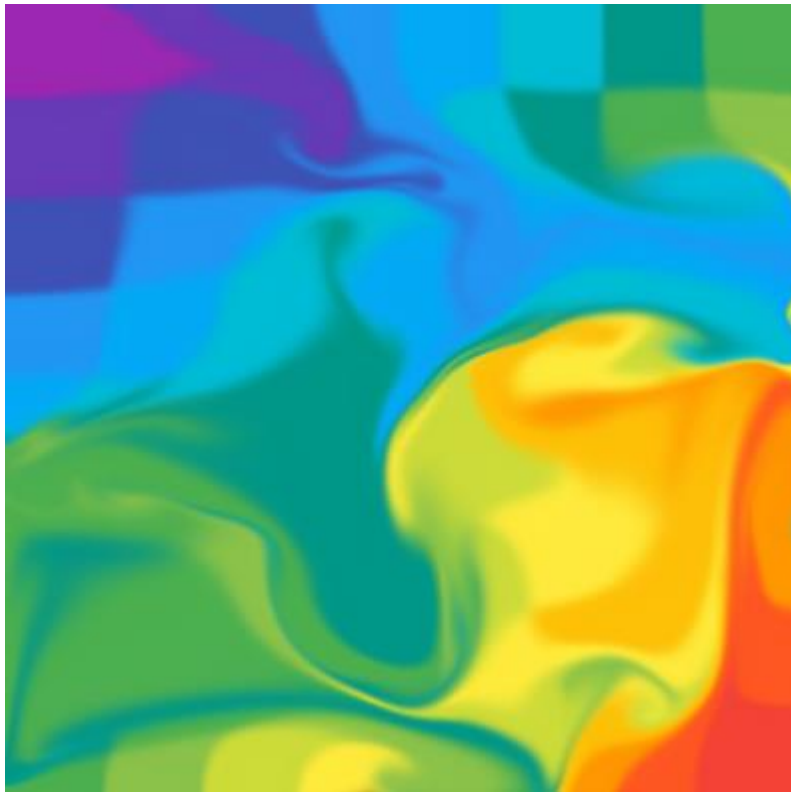
Implement Pipeline

1. Compute “advect”, “diffusion”, and “force” to get w
2. Apply divergence operator to solve gradient of p
3. Finally, w subtrats gradinet p to get u

Experiment result: show fluid velocity as color digree



Experiment: apply velocity on texture



Parallelism

Each step can be compute independently on each pixels, just allocate the work to each cuda core. We use `cudaMallocManaged()` to manage GPU memory manage.

Performance and improve

Environment:

Windows 10

Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz

Nvidia GTX1050

Result:

with only one CPU core: 10+-fps, 150 * 150 will reach max CPU load

with GPU : 100+-fps, 400* 400 run smoothly

用 CPU 去運行，以上配置便接近極限，而使用 GPU 可以再 400*400 pixels 運算量七倍以上的圖片上，FPS 有 10 倍的提升，且還很順。

improve:

我們認為目前最大的限制在於我們使用 SFML 來進行視窗顯示與滑鼠等操作，若是直接使用底層的 OpenGL 來進行繪圖，應該能達到更好的效能，或是 improve fluid 的演算法。