Homework #02: Polygon Fountain

Computer Graphics
School of Software
College of Computer Science, Kookmin University

Abstract

In this assignment, you create a virtual 2D fountain emitting a lot of polygons, called *polygon fountain*, such as Fig. 1.

✦

폴리곤은 삼각,사각 상관없는디 예쁘면 추가점수
제대로 구현됐는지... 색을 예쁘게 배치???, 이펙트 추가되면 추가점수 존재

폴리곤으로 분수 만들기

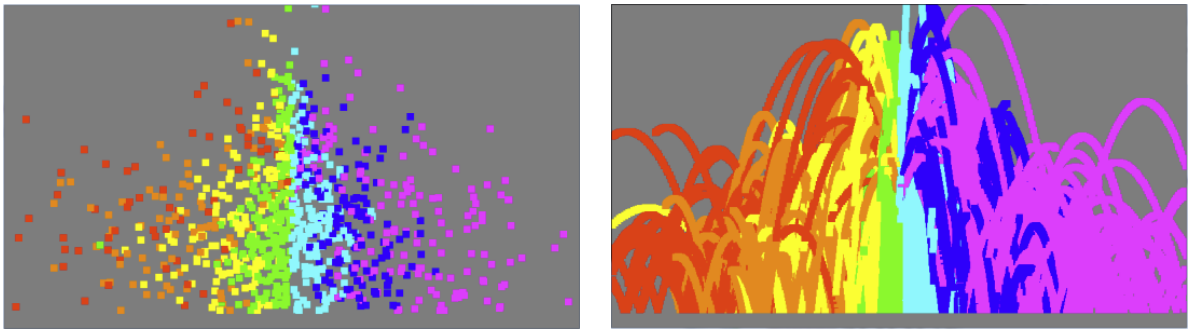

Fig. 1. Examples of polygon fountains

폴리곤이 화면밖으로 나가면 그걸 다시 주워와서
다시써도되고, 재생성해도 된다.
막 느려져도 안된다(최적화?)

1 Introduction

Suppose the emitter of the polygon fountain located in the 2D position $P_0 = (x_0, y_0)$. When the emitter emits a polygon with the initial velocity $v = (v_x, v_y)$, the following is the location of the polygon at time t:

각각의 폴리곤들은 속력/중력에 영향을 받아 분수 형태를 이룬다

$$P_t = \left(x_0 + v_x t,\ \ y_0 + v_y t - \frac{1}{2} g t^2 \right), \tag{1}$$

where g is the gravity constant (i.e., $9.8 m/s^2$).

For each polygon, you may change the initial velocity v, by using random functions. The goal of this homework is to generate your own polygonal fountain in an artistic way. Your program must satisfy the following criteria, where you can freely add up your own criteria in the program.

1) Your program visualizes a part of the scene with the polygonal fountain. Initially, the part of the scene should be bounded by $x \in (-5, 5)$ and $y \in (0, 10)$. x축으론 -5 to 5 , y축으론 0 to 10 사이에서 보여주면 된다.
2) You can change your camera settings by pressing the following keys.
   - '←' key: camera moves to $-x$ direction.
   - '→' key: camera moves to $+x$ direction.
   - '↑' key: camera moves to $+y$ direction.
   - '↓' key: camera moves to $-y$ the down direction
   - '+' key: camera zoom-ins the scene.
   - '–' key: camera zoom-outs the scene.
   - 'Home' key: camera recovers the initial settings.

   방향키를 이용해서 좌우아래위, 플마키를 이용해서 확대축소가 가능해야 한다.
   홈키를 누르면 카메라가 최초 세팅으로 돌아간다.
3) In your program, you can control the number of polygons emitted from the polygonal fountain, with the context menu.
   - # of polygons = 10
   - # of polygons = 50
   - # of polygons = 100
   - # of polygons = 500
   - # of polygons = 1000

   폴리곤 갯수를 조절할 수 있어야 한다.
   10~1000 (freeglut활용)
   컨텍스트 메뉴를 사용

Notice that you visualize only a part of the scene with the polygonal fountain, in general. As a result, you must develop your own way to optimize the program by using the fact that your program only visualizes a part of the scene.

국민대학교 소프트웨어학부 김준호, junho@kookmin.ac.kr

폴리곤이 이동하면서 히오스마냥 빙빙돌면?

2 What you learn

From this homework you can learn about the followings

- How to visualize a scene with Modern OpenGL
  - Write a your own shader program consisting of a vertex shader and a fragment shader
  - Use Modern OpenGL functions, related to drawing polygons with shaders, such as
    - glUseProgram
    - glEnableVertexAttribArray, glDisableVertexAttribArray
    - glVertexAttribPointer,
    - glDrawArrays

GlDrawArrays
모델매트릭스 사용해서 폴리곤 이동

- How to transform an object in the scene wity a model matrix, by using your transformation functions.
- How to manipulate a your virtual camera wity a view matrix, by using your own looAt() function.
- How to update window displays with FreeGLUT.
  - Use glutDisplayFunc, glutIdleFunc, glutIdleFunc

디스플레이 펑션은 화면에 그림만 그려야 하고
그림(모델)위치를 이동시키는건 idle펑션에서 다뤄야 한다.
(디스플레이 펑션에 부하가 강해지면 렉이 걸림)

- How to use keyboard callback functions with FreeGLUT.
  - Use glutKeyboardFunc, glutSpecialFunc

glut스페셜펑션 이용안하면 키보드 충돌할수도 있음

- How to make a menu with FreeGLUT.
  - Use glutCreateMenu, glutAddMenuEntry, glutAttachMenu

팀원 중 한명만 가상대학에 올리며ㄴ된다

3 Deadline and Misc. (EXTREMELY IMPORTANT!!!)

- This assignment can be accomplished in a team, typically composed with 1 or 2 people.
  - Caution 1: Once you have a team, you cannot change it until the end of the semester.
  - Caution 2: Only one of your team members submits your homework.
- Your homework must be submitted to 가상대학, until 23:59 on April. 28.
- You should compress the followings into a **tar.gz** or **zip** file whose name is OOOOOOOO_HW02.tar.gz (OOOOOOOO should be your student ID).

컴파일 된 파일은 넣지말고
소스랑 메이크파일까지만 넣.

  1) Your source files
  2) Makefile
  3) README.txt file, including the student numbers of the team members.
- **Your source codes must be complied with a make command in Ubuntu 16.04 LTS.**
- If you ask some questions to me, please utilize the office hours. It is also fine that you contact T.A. for asking some questions about the homework.