**Project1-Dolly Zoom**

一、运行环境

操作系统：Windows11

Python版本：3.8

第三方库：无

二、过程分析

①compute\_focal\_length：

给定感兴趣的物体的深度d\_ref以及摄像机的初始焦距f\_ref，计算更新后的焦距f，使得物体在沿着z轴运动pos之后物体的高度保持不变。

由已知条件可得：

故：

②compute\_f\_pos：

给定感兴趣的两个物体，它们的深度分别为d1\_ref和d2\_ref，以及它们在三维物理世界的高度分别为H1和H2，摄像机的初始焦距为f\_ref。计算更新后的焦距f以及相机的新位置pos，使得第一个物体的高度在图中保持不变，且两个物体在图像中的高度比为ratio=h1/h2。

由

可得：

故：

同①可得：

三、运行结果

①compute\_focal\_length：

d\_ref = 4，f\_ref = 400，pos = -5

则：f=900

②compute\_f\_pos：

d1\_ref = 4，d2\_ref = 20，H1 = 4，H2 = 6，ratio = 2，f\_ref = 400

则：f=800，pos=-4

四、源代码

|  |
| --- |
| **def** compute\_focal\_length(d\_ref, f\_ref, pos):  *"""  compute camera focal length using given camera position   Input:  - d\_ref: 1 by 1 double, distance of the object whose size remains constant  - f\_ref: 1 by 1 double, previous camera focal length  - pos: 1 by n, each element represent camera center position on the z axis.  Output:  - f: 1 by n, camera focal length  """   # IMPLEMENT HERE* f = f\_ref \* (d\_ref - pos) / d\_ref   **return** f   **''' Test '''** d\_ref = 4 f\_ref = 400 pos = -5 f = compute\_focal\_length(d\_ref, f\_ref, pos) print(f)   **def** compute\_f\_pos(d1\_ref, d2\_ref, H1, H2, ratio, f\_ref):  *"""  compute camera focal length and camera position to achieve certain ratio  between objects   Input:  - d1\_ref: distance of the first object  - d2\_ref: distance of the second object  - H1: height of the first object in physical world  - H2: height of the second object in physical world  - ratio: ratio between two objects (h1/h2)  - f\_ref: 1 by 1 double, previous camera focal length  Output:  - f: 1 by 1, camera focal length  - pos: 1 by 1, camera position  """   # IMPLEMENT HERE* pos = (H1 \* d2\_ref - ratio \* H2 \* d1\_ref) / (H1 - ratio \* H2)  f = f\_ref \* (d1\_ref - pos) / d1\_ref   **return** f, pos   **''' Test '''** d1\_ref = 4 d2\_ref = 20 H1 = 4 H2 = 6 ratio = 2 f\_ref = 400 f, pos = compute\_f\_pos(d1\_ref, d2\_ref, H1, H2, ratio, f\_ref) print(f, pos) |