Team Control Number

22875633

Problem Chosen

D

2022

IMMC

Summary Sheet

Building a virtual Hong Kong in metaverse

Team#22875633

January 22, 2022

Contents

1 Introduction

1.1 Background

Nowadays, *metaverse* has become a heated topic and a symbol of development in virtual technology. It refers to a three-dimensional virtual world, especially in online role-playing games such as *Second Life*, *Minecraft* and *Roblox*. In this system, people can build virtual items according to their own needs and complete basic communication with others such as buying and selling items from others as well as sharing information of their creations. In other words, this is a society which only depends on the instructions of game players. Using this technology, we can simulate the society by uploading the data given and create the items which exist in daily life to predict the future development of this system.

In the three games mentioned above, players use limited basic blocks and their own imagination to build their own virtual world. It can be seen that the type of basic blocks is very important. They must meet the needs of construction to the greatest extent, and at the same time, the number of types and size should be appropriate so that players can easily understand the function of each blocks and make full use of them.

Now Hong Kong is also confronted with the problem of how to improve its urban construction without imposing policies on the real city. On this occasion, creating a virtual Hong Kong by using metaverse is key to this challenge. We can create simple elements in this virtual world which fit the reality so that government can observe the effect of their policies by giving basic instructions in this virtual city. In this way, there will be no loss in the real world due to strategic mistakes because all of the mistakes actually happen in real life. City construction in Hong Kong can be easier and more efficient.

1.2 Problem Restatement

To build a virtual city and make sure that it can be used in daily thing, we need to figure out its basic elements and find out the computing and storage resources needed to maintain a virtual Hong Kong. So our work is divided into 2 parts:

- Build the basic blocks that satisfy the needs of citizens and can be seen in a real city. We will explain the functions of each block and how it can be planted in the virtual city.
- Calculate the fitting degree of basic blocks to make sure our virtual city can simulate Hong Kong.
- Describe the computing and storage resources required in our design by calculating the speed at which it update the state of moving objects(such as pedestrians and cars) and how much bytes of storage it needs to maintain a virtual city.

1.3 General Assumptions

- **The ground of Hong Kong can be regarded as a plain.**

  Though we all know the earth is an ellipsoid, Hong Kong is so small that we can regard its ground as a plain. Also, the buildings in Hong Kong are generally not high. So we can neglect factors of ground inclination.

- **The data will be updated non automatically when serious natural disasters such as typhoons and hurricanes destroy buildings.**

2 Model A:Determining the Basic Blocks

2.1 Model Overview

In this model, we will mainly discuss the basic block sets in this society. We divide the blocks into two parts: how to describe the location and features of the blocks and how to make sure our block fit the constructions in real Hong Kong. There are two kinds of blocks in this model: blocks which have shapes and blocks which have entity. Later we will discuss the relationship between them.

When we take everyday life into consideration, we will find that what we need to build is divided into two parts:

- **Transportation vehicles**
  In Hong Kong, there are four types of transport: rail transit, air transport, transport on sea and road transport.

- **Buildings**
  These buildings consists of building blocks and are seen as static objects.

In this way, we can build a basic structure of our virtual city.

2.2 Notation

f(block)	properties of the block
V	volume of the block
ρ_1	continuity of lines in patterns
ρ_2	density of lines in patterns
L	dimension of lines in patterns
α_{radius}	a certain proportion which determines L
r	radius of the dots in patterns
θ_1	angle with O_x
θ_2	angle with O_y
k	a special property of each block

2.3 Creating the Blocks

In our model, we use a coordinate system to describe the absolute location of each blocks which can enable us to quickly track the route of each moving block. Also, we use a matrix to describe the key properties which will be introduced later.

We use a shape set to describe the geometric features of blocks. There are several kinds of geometry.

Type 0 is hexahedron, which is shown in the picture below. α, β, γ, a, b, c, θ_1, θ_2 can be introduced to describe its shape. Also,k is introduced to describe its dimension. It means that, when k=0, it's a pyramid while when k=1, it's a parallelepiped.

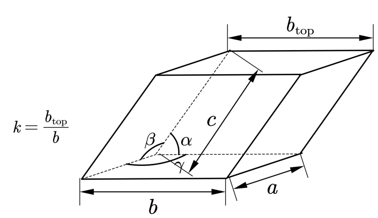

Fig.3 An example of parallelepiped

Type 1 is ellipsoid. a, b, c is introduced to determine its shape.

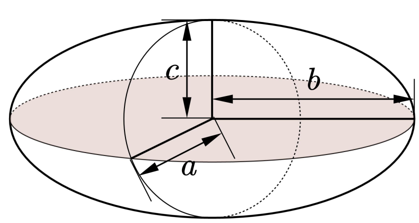

Fig.4 An example of ellipsoid

Type 2 is elliptic column platform. This time, a, b, h, α, β, k is introduced to describe its shape.

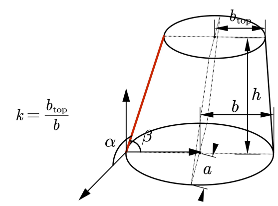

Fig. 5 An example of elliptic column platform

Secondly, we need to describe the color and patterns on the blocks. We use three primary colors(red, blue and yellow) to describe the color of blocks. For most of the patterns are made up of lines, we use ρ_1 and ρ_2 to describe the features of these patterns.

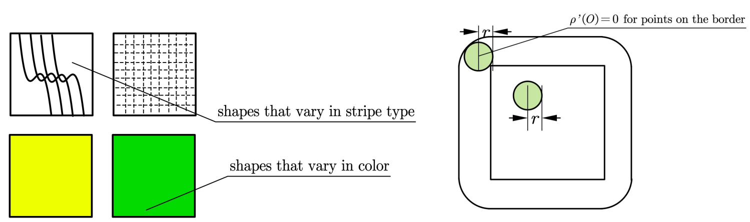

Fig.1&2 Patterns on the blocks

According to the picture, we can know the correct definition of r and L. We use α_{radius} to determine the proportion of this dimension.

$$r \overset{def}{=\!=} L \times \alpha_{radius} \tag{2.1}$$

So, when we draw several circles with O as the center(O refers to every spot in the surface), we can find that:

$$\rho'(o) = \frac{\text{Line Area}}{\pi r^2} \tag{2.2}$$

They can be defined as the formulae below.

$$\rho_1 \overset{def}{=\!=} \frac{\max\limits_{\text{Surface}} \rho'(o) - \min\limits_{\text{Surface}} \rho'(o)}{L} \tag{2.3}$$

$$\rho_2 \overset{def}{=\!=} \frac{\iint_{Surface} \rho'(x, y) d\sigma}{S(Surface)} \tag{2.4}$$

So, when we combine all the properties together, we can get the matrix:

$$\text{f(block)}=\begin{pmatrix} \text{V} & & & \\ & (\rho_1, \rho_2) & & \\ & & (\text{R}, \text{G}, \text{B}) & \\ & & & \text{vectors in the shape set} \end{pmatrix}$$

(In this part, vectors refer to their determining factors of the block's shape)

3 Calculating the Fitting Degree

3.1 Problem Overview

In this part, we will discuss the fitting degree of our blocks. Calculations will be focused on its shape. Blocks will be divided into different parts and using computing technology, we can compare basic properties mentioned it the last part with items in reality. In this part, Q-learning is introduced to facilitate calculations.

3.2 Notation

F_1	the fitting degree in shape
F_2	the fitting degree in lines
F_3	the fitting degree in color
$Overlap_{Cluster}$	the degree in which blocks and items overlap
$Excessive_{Cluster}$	the degree in which blocks vary with items
$\Delta\rho_1$	change in continuity
$\Delta\rho_2$	change in density
Δ_e	the maximum color difference distinguishable by human eyes
Δ_{color}	color difference of blocks

3.3 Simulating the Shape

Firstly, we need to calculate the fitting degree of the block's shape. We separate the degree into two parts:$Overlap_{Cluster}$ and $Excessive_{Cluster}$. So we can the fitting degree shape can be described as:

$$F_1 = \frac{Overlap_{Cluster} - Excessive_{Cluster}}{V} \tag{3.1}$$

Secondly, we will discuss the fitting degree in patterns on blocks. It will be divided into lines, colors and shape. $\Delta\rho_1$ and $\Delta\rho_2$ is introduced to explain the relationship between blocks and items. So we can get the following formulae:

$$\Delta\rho_1 = |\rho_1(surface(Block)) - \rho_1(surface(Target))| \tag{3.2}$$
$$\Delta\rho_2 = |\rho_2(surface(Block)) - \rho_2(surface(Target))| \tag{3.3}$$
$$F_2 = e^{-(k_1\Delta\rho_1 + k_2\Delta\rho_2)} \tag{3.4}$$

When we discuss the fitting degree in color, we identify Δ_e as the maximum color difference distinguishable by human eyes. So when we compare $(R_Block, G_Block, B_Block)$ with $(R_Target, G_Target, B_Target)$, it is clear that:

$$\Delta_{color} \overset{def}{=\!=} \sqrt{\sum_{R,G,B} \frac{\left(R_{Block} - R_{Target}\right)^2}{3}} \tag{3.5}$$
$$F_3 = e^{1-\frac{\Delta_{color}}{\Delta_e}} \tag{3.6}$$

(When $\Delta_{color} = \Delta_e$, the fitting degree is 100%.)

3.4 Calculating the Traffic Flow

As we need to make the virtual Hong Kong seem as real as possible, we need some data provided by the government to make the virtual Hong Kong look like Hong Kong in real life.

We need the information of the buildings in Hong Kong, as we need to place the blocks in the virtual Hong Kong. Meanwhile, we need the data about Hong Kong’s population and transportation to estimate how much calculation the virtual Hong Kong needs. This kind of data include data of Hong Kong’s population, total amount of cars, total traffic volume and total area. These data can be gotten from the government. In addition, we need the data of human eye resolution to confirm an adequate resolution for the virtual Hong Kong. Fortunately, all the data mentioned above can be found in the Internet.

备用的伪代码

Algorithm 1 Q-Learning 2

divide the whole area into several cubic parts
for part (i, j, k): # simulate the single part **do**
  initialize Q table # the potential benefits of the plan, a matrix
  initialize **S** # the set of actions to simulate the city with geometric forms, a list
  initialize and # is the rate of learning while is the rate of declining. Both and are in the range of (0, 1).
  choose action **A** from **S** due to a policy from **Q** # **A** is a plan of fitting the blocks with geometric forms, choose the plan with the highest simulation level
  while not done: **do**
    take action **A**, observe **R, S'**
    Q(S,A)←Q(S,A)+[R+maxa Q(S ',a)Q(S,A)]# update the Q table and the status S ←S'
  end while
end for
for the borders of parts: **do**
  small adjustions to simulate the area better
end for

References

[1] EPSG:2326 Hong Kong 1980 Grid System, MapTiler Team
https://epsg.io/2326