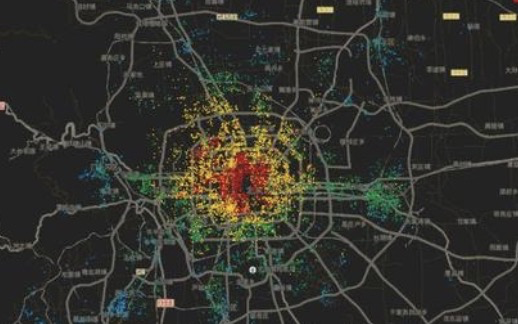
# 3D热力效果开发的流程记录

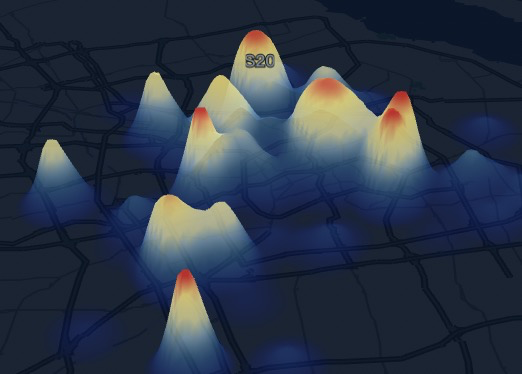
热力效果图能以高亮的形式显示区域活动的密集程度，能让用户直观的感受区域活动的轨迹，为用户的决策提供参考。

传统的热力图多数以平面热力图为多，例如：





但是随着页面3D的不断普及，以及3D效果能给用户呈现出更好的视觉效果，3D热力图也开始成为主要的热力呈现手段，如高德地图中和3D热力：



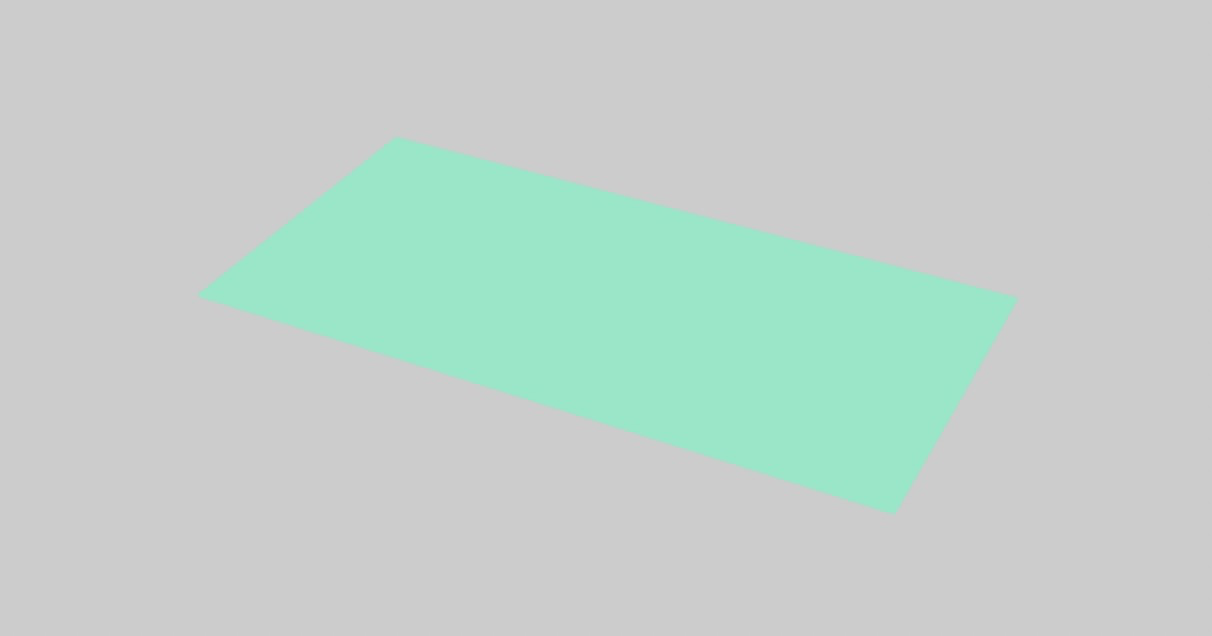
在研究开发3D热力图的过程中，其中重要的是表示热力信息的3D坡面的生成。

在本文中3D 对象的开发使用THREE.js

生成3D坡面的基本做法的思路如下：

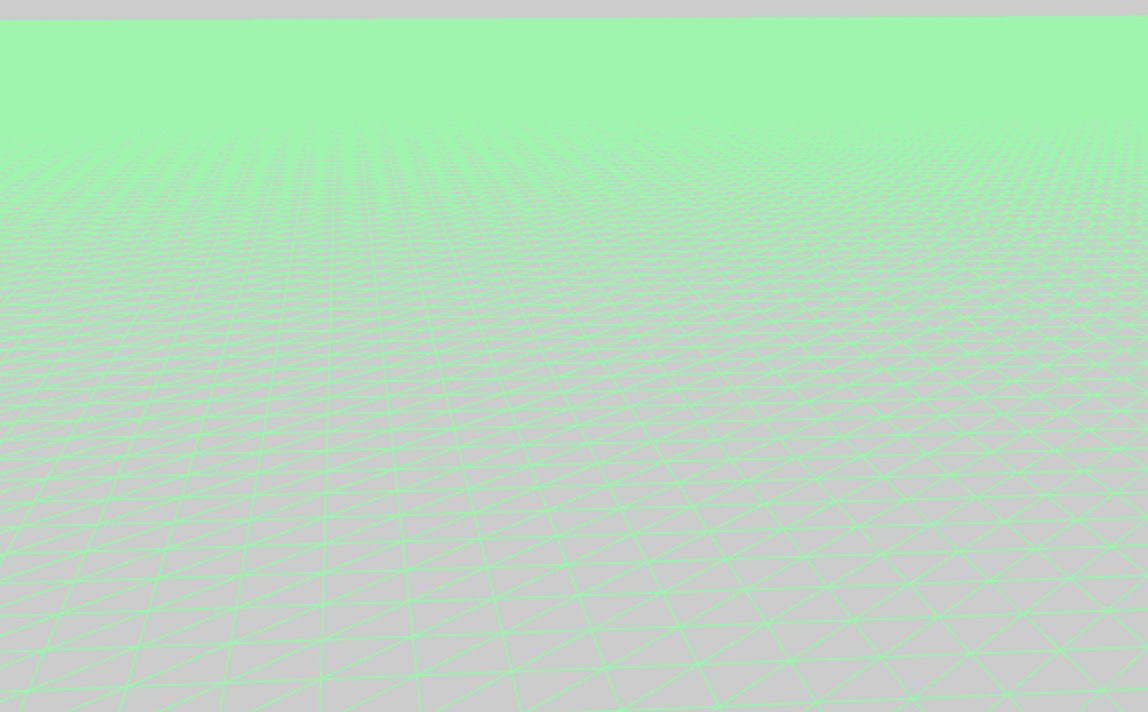
1. 首先生成作为基面的平面

使用 THREE.PlaneGeometry 对象生成基础的平面



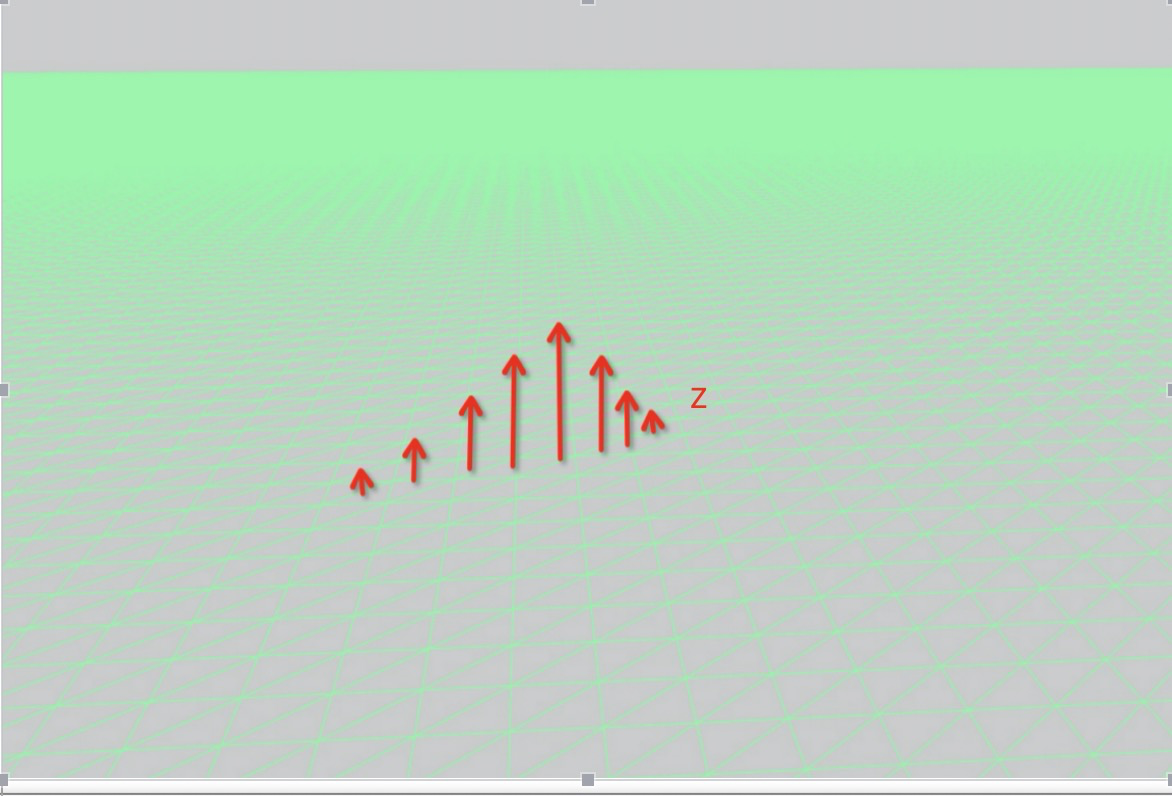
1. 确定平面上每个节点

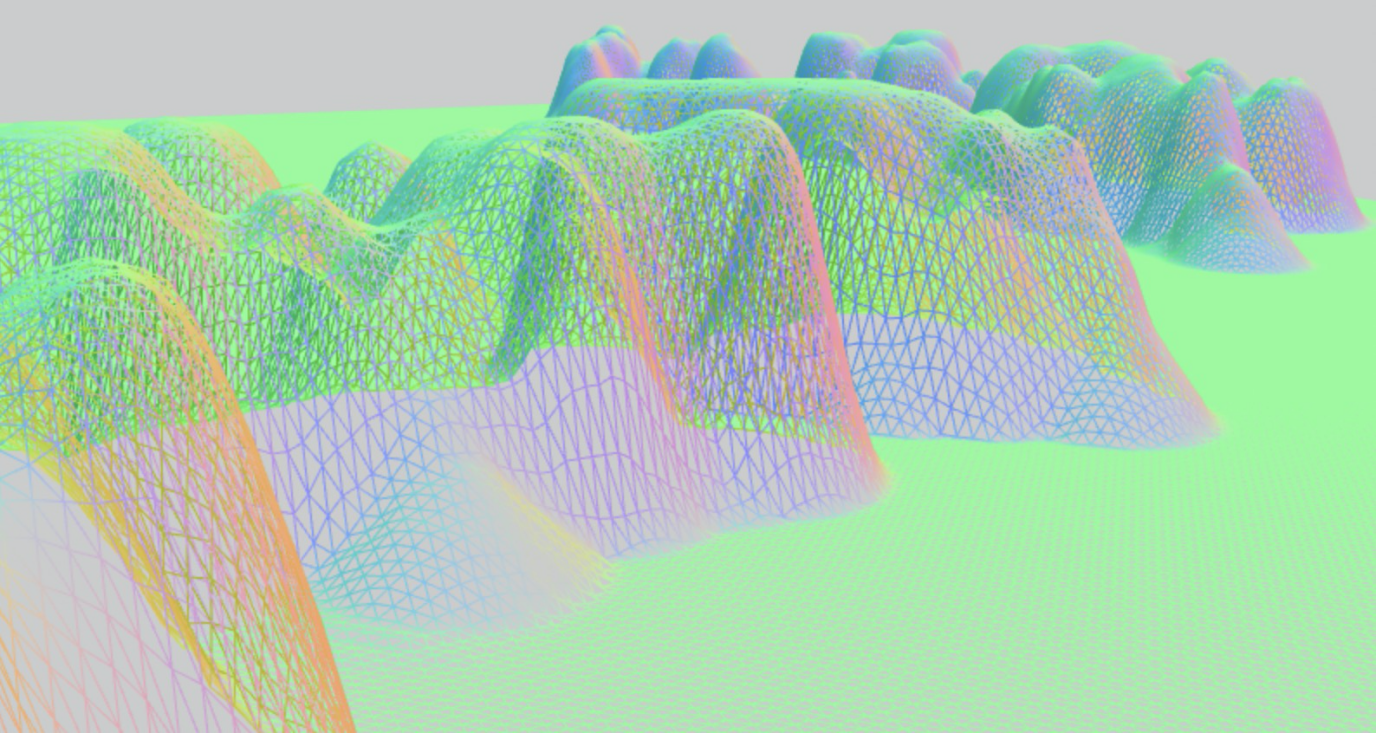
使用 geometry 网格的 vertices 对象保存网格对象的节点信息



1. 根据热力信息为每个节点赋予坡面的高度信息

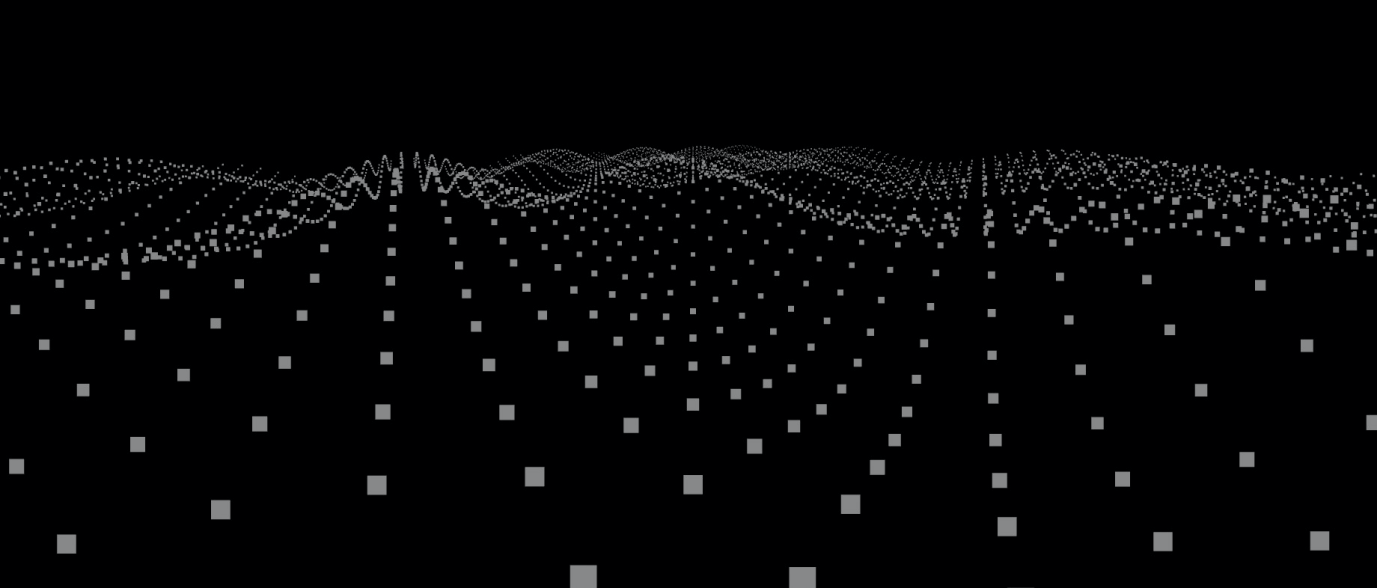
此时坡面的高度信息是节点分量是vertices的z 分量

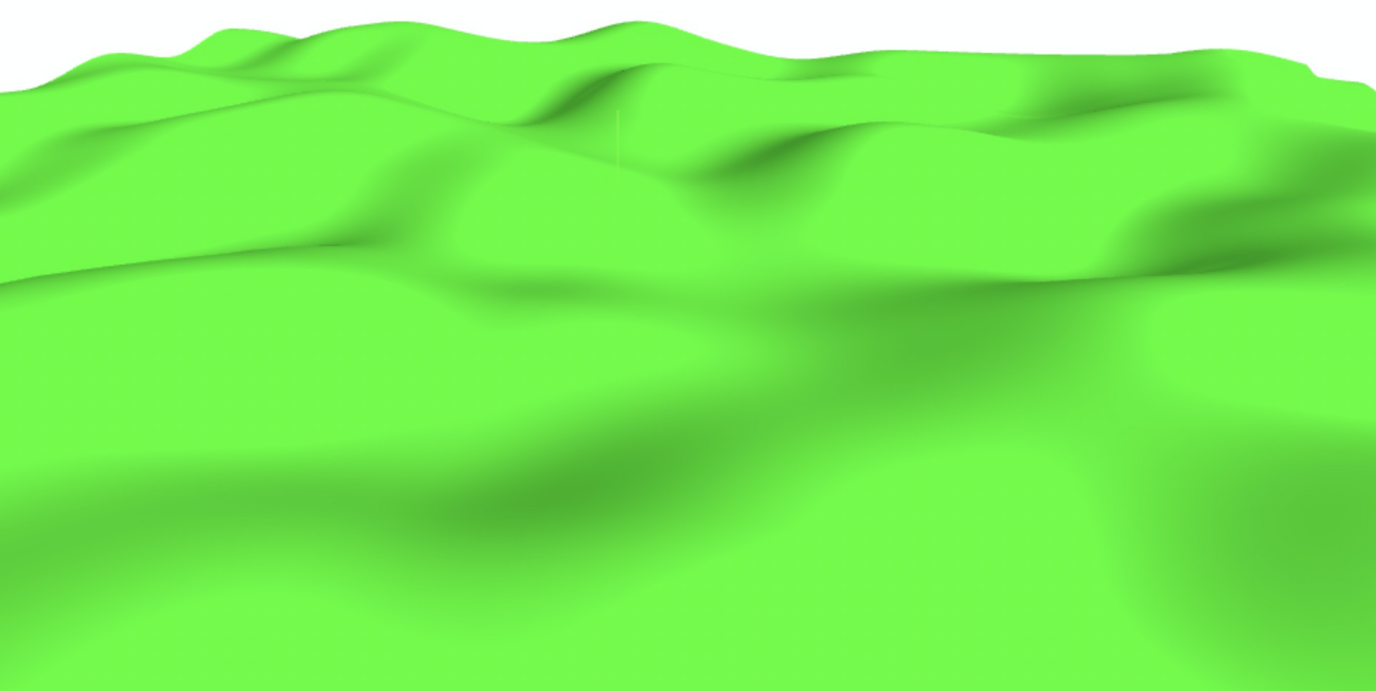




如何为每个节点生成正确的高度信息是关键

刚开始的思路是先给出的几个位置的高度信息，并使用数学方法生成坡面其他位置的高度信息





就像这两个例子一样，分别采用三角函数推导和柏林噪声算法来计算每个顶点的高度。然而在实际开发的过程中发现，由于热力信息并没有规律性，导致使用数学计算生成坡面高度的实施十分困难，这时候开始调整思路。

在与同事讨论的过程中，发现我们可以在现有十分成熟的2D热力图中提取高度信息，原理是根据热力颜色的渐变间接的获取坡面高度信息。

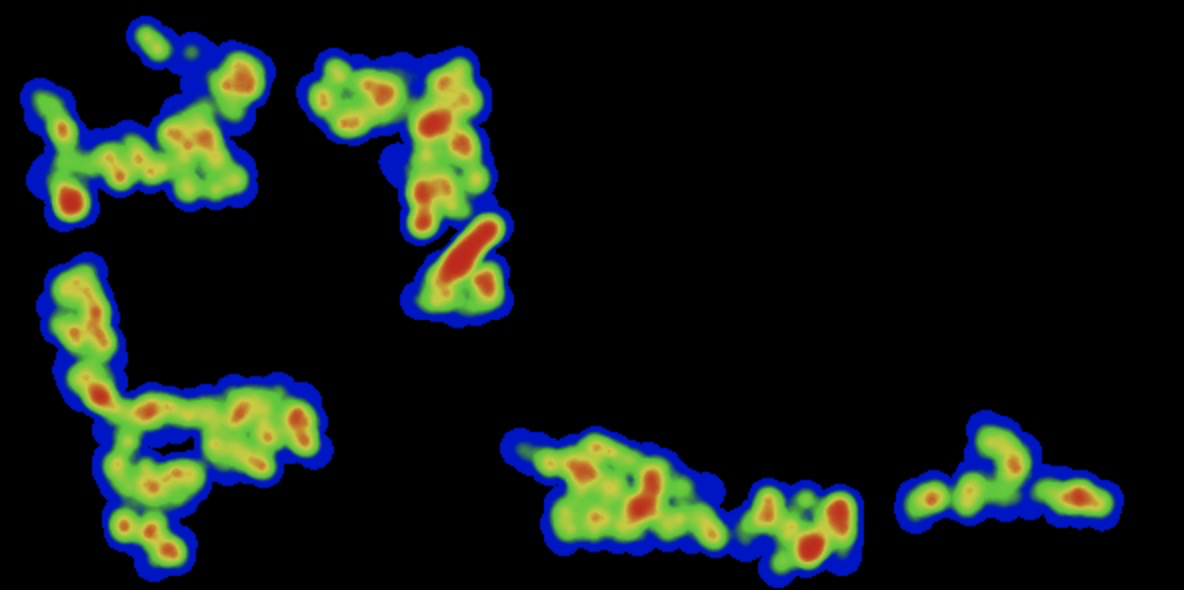
( 本文中2D热力的生成使用heatmap.js )





如图，在只给定一个点的数据的情况下，颜色的渐变十分明显，我们可以使用canvas的 getImageData 的方来来获取像素的颜色值，并同时将热力的颜色设置成单色，方便数值的计算。

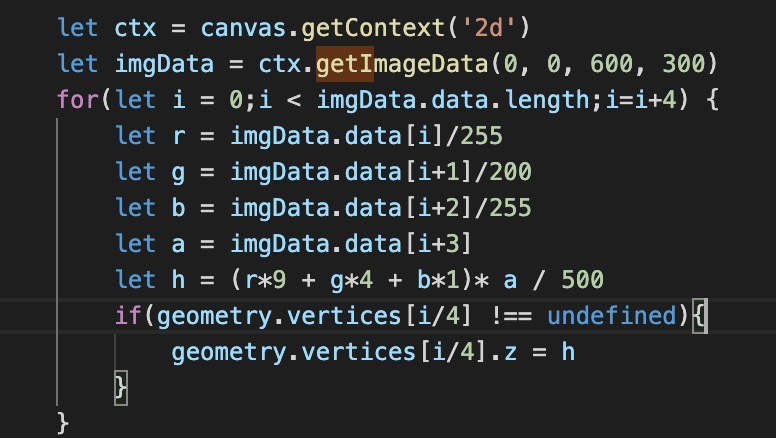
原图热力效果



单色渐变热力效果



使用canvas提取渐变颜色的具体代码如下：



在这里需要注意的是如何将颜色取值与网格对象的每一个节点一一对应，首先需要保证的颜色的取值数和节点的数一致。

canvas颜色取值为 600\*300 = 180000 个颜色值

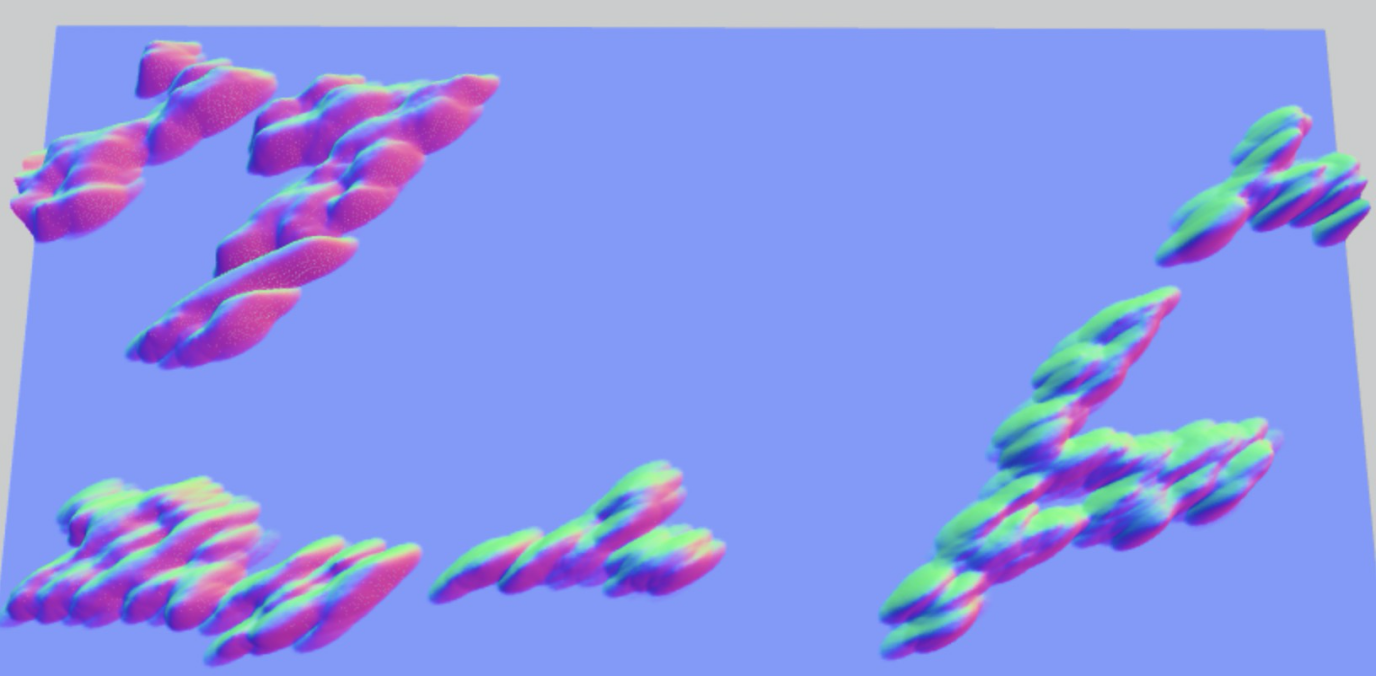


在构建网格对象的时候采取长宽分别是600 和 300 的分段



但此时实际上网格的节点数是 180901 个节点

此时发生节点对应不上的情况

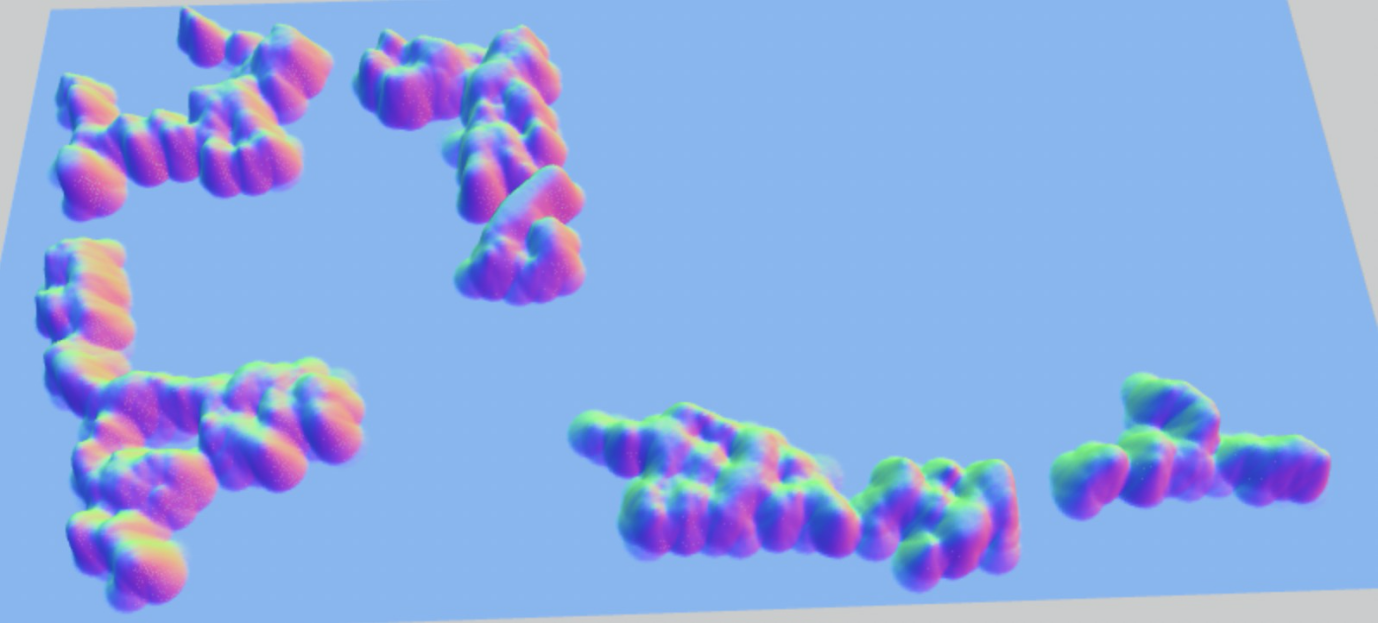


此时应该将长宽的分段数同时减一



此是网格的节点数才是 600\*900 = 180000

这是节点对应正确匹配的情况

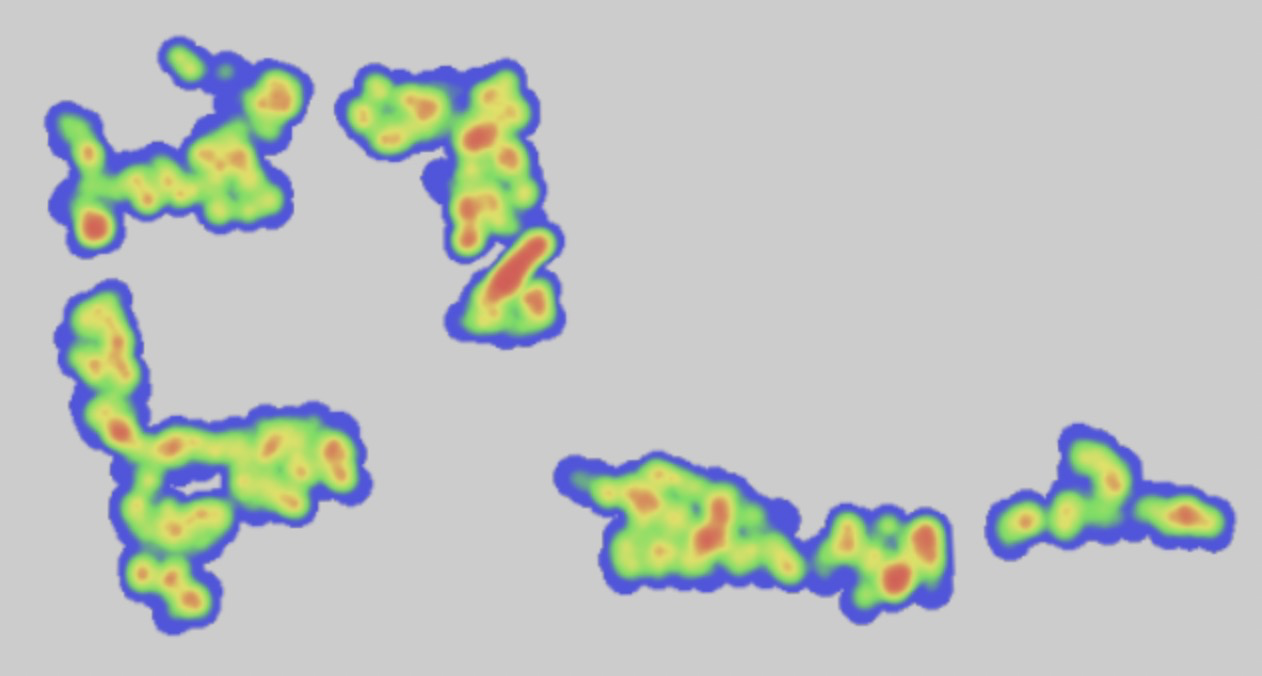


可以明显发现坡面有偏移的现象

在生成 3D 坡面之后需要为坡面赋予热力材质。

为坡面赋予材质的方法与普通的网格对象贴图操作一致

这是平面热力的效果



这是3D 的效果

