生成点云模板

- STEP 1：测量模板尺寸

Number	length	width	height	unit
# 1	140	130	35	mm
# 2	167	125	60	mm
# 3	145	85	105	mm
# 4	170	93	112	mm
# 5	195	107	135	mm
# 6	133	80	90	mm
# 7	150	85	110	mm
# 8	145	90	110	mm
# 9	115	88	103	mm
# 10	195	105	135	mm

- STEP 2： Matlab生成模板坐标

```
function generate_pointcloud_position(X,Y,Z,step,save_path)
%% 生成点云坐标
%step:步长

%X surface
disp("processing on generating X surface data...");
p=[0 0 0];
for x=[0 X]
    for y=0:step:Y
        for z=0:step:Z
            p=[p;x,y,z];
        end
    end
end
% Y surface
disp("processing on generating Y surface data...");
for y=[0 Y]
    for x=0:step:Y
        for z=0:step:Z
            p=[p;x,y,z];
        end
    end
```

```
end
% Z surface
disp("processing on generating Z surface data...");
for z=[0 Z]
    for x=0:step:X
        for y=0:step:Y
            p=[p;x,y,z];
        end
    end
end
disp("saving data...");
% save
dlmwrite(save_path,p);
disp("Done!");
```

- STEP 3: 使用Python转换格式

```
for i in range(10):
    print(i)
    origin_path="./#%d.txt"%(i+1)
    save_path="./#%d_rgb.txt"%(i+1)

    data=[]
    with open(origin_path,"r") as f:
        lines=f.readlines()
        for line in lines:
            line=line.replace(","," ")
            line=line.split("\n")[0]+' 255 255 255\n'
            data.append(line)
    with open(save_path,'w')as f:
        for line in data:
            f.write(line)
```

- STEP 4 :使用PCL生成pcd点云

```
#include <iostream>
#include <fstream>
// PCL 库
#include <pcl/io/pcd_io.h>
#include <pcl/point_types.h>
#include <pcl/io/ply_io.h>
#include <pcl/visualization/cloud_viewer.h>
// 定义点云类型
typedef pcl::PointXYZRGB PointT;
typedef pcl::PointCloud<PointT> PointCloud;

using namespace std;

int main(int argc,char* argv[])
{   if(argc<2){
      cerr<<sizeof(argc)/sizeof(int)<<2<<" less paras to run..."<<endl;
      return -1;
    }
    // 点云变量
    // 使用智能指针，创建一个空点云。这种指针用完会自动释放。
    fstream infile;
    infile.open(argv[1], ios::in);
    if (!infile.is_open()){
      cerr<<"open file error."<<endl;
      return -1;
    }
    PointCloud::Ptr cloud(new PointCloud);
    float x, y, z;
    int r, g, b;
    while (!infile.eof())
    {
        infile >> x >> y >> z >> r >> g >> b;
        // d 存在值，则向点云增加一个点
        PointT p;
        p.x = x;
        p.y = y;
        p.z = z;
        p.r = r;
        p.g = g;
        p.b = b;
        // 把p加入到点云中
        cloud->points.push_back(p);
    }
    infile.close();
    // 设置并保存点云
    cloud->height = 1;
    cloud->width = cloud->points.size();
    cout <<"  point cloud size = " << cloud->points.size() << endl;
    cloud->is_dense = false;
    pcl::io::savePCDFile(argv[2], *cloud);
    /*pcl::visualization::CloudViewer viewer("Viewer");
    viewer.showCloud(cloud);*/
    // 清除数据并退出
    cloud->points.clear();
    cout << "Point cloud saved." << endl;
    return 0;
}
```

- 查看结果

```
pcl_viewer exmaple.pcd
```