Stratified Mesh Sampling for VTK

Release 0.00

David Doria

March 12, 2010

Rensselaer Polytechnic Institute, Troy NY

Abstract

This document presents an implementation of a stratified mesh sampling algorithm in the VTK framework. This code was adapted directly from the original implementation by Diego Nehab [1], with permission. We present a class, *vtkStratifiedSampling*, which uniformly samples a mesh. A Paraview plugin interface is provided to allow extremely easy experimentation with the new functionality. We propose these classes as an addition to the Visualization Toolkit.

Latest version available at the Insight Journal [http://hdl.handle.net/10380/3156]

Contents

1 Introduction

It is often desired to obtain a point cloud of uniformly spaced points on a mesh. This paper presents the implementation of an algorithm for producing such a set of points. This method was originally published in

[1], we have simply implemented it in the VTK framework.

2 vtkStratifiedSampling

2.1 Options

Level

This value controls the density of the points. The default value is 6. The higher the value, the more points are produced. This value can be set using:

```
stratifiedSampling->SetLevel(10);
```

Lambda

This values specifies the parameter of the exponential probability distribution from which to sample the points. The default value is 10.0.

This value can be set using:

```
stratifiedSampling->SetLambda(1.1);
```

Bad

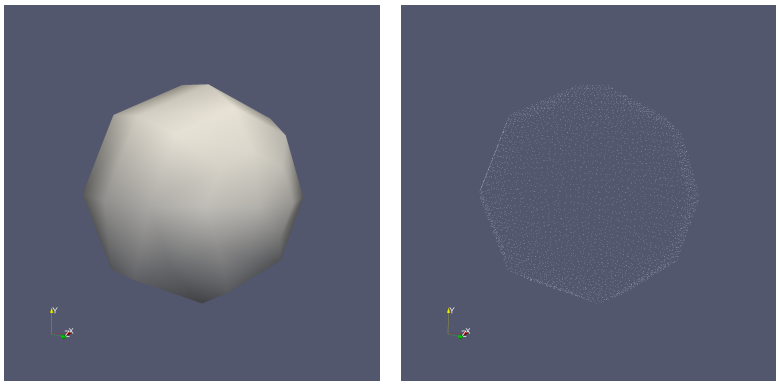
A threshold to remove points that are too close to each other. The default value is 0.5.

This value can be set using:

```
stratifiedSampling->SetBad(.5);
```

2.2 Demonstration

To demonstrate stratified sampling, we use a sphere. This data set is shown in Figure 1(a). In Figure 1(b), we show the stratified sampling.



(a) Sphere (b) Stratified Sphere Sampling

Figure 1: Stratified sampling demonstration.

2.3 Code Snippet

```
vtkSmartPointer<vtkSphereSource> sphereSource =
    vtkSmartPointer<vtkSphereSource>::New();
sphereSource->Update();

vtkSmartPointer<vtkStratifiedSampling> stratifiedSampling =
    vtkSmartPointer<vtkStratifiedSampling>::New();
stratifiedSampling->SetInputConnection(sphereSource->GetOutputPort());
stratifiedSampling->Update();

vtkSmartPointer<vtkXMLPolyDataWriter> writer =
    vtkSmartPointer<vtkXMLPolyDataWriter>::New();
writer->SetInputConnection(stratifiedSampling->GetOutputPort());
writer->SetFileName("test.vtp");
writer->Write();
```

3 Paraview Plugin

For convenience, this code is shipped with a Paraview filter plugin. The plugin provides an easy way to set the parameters as well as integrate the new code into your workflow. A screenshot of the plugin interface is shown in Figure 3.

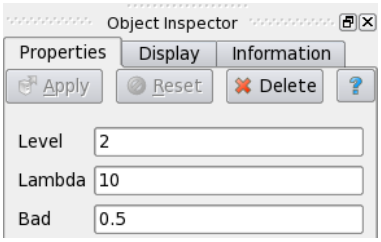


Figure 2: Paraview plugin screenshot

References

[1] D. Nehab and P. Shilane. Stratified point sampling of 3d models. *Eurographics Symposium on Point-Based Graphics*, 2004. (document), 1