

Design and develop the 3D seismic data visualization software based on VTK and QT

Jie Xiong¹, Chenjing Xia¹, Shenhan Shi¹

¹School of Electronics and Information, Yangtze University, Jingzhou, China.

Abstract:

Seismic exploration is an important method for the oil-gas exploration. The visualization of 3D seismic data can help geophysicists to find the oil-gas resources more effective. In order to develop a visualization software which can run on the Windows and Linux, and rendering the 3D data effectively, we choose the QT framework and Visualization Toolkit (VTK). The source code is managed by the CMake, which helps to migrate the software from Windows to Linux smoothly. We choose the ray casting algorithm to accelerate the 3D volume rendering. The software is developed successfully, and tested. The test result shows the QT framework and VTK can be used to develop a 3D visualization software with the characteristic of cross-platform.

Keywords

3D Visualization; Seismic Data; QT; Visualization Toolkit (VTK); ray casting algorithm.

1. Introduction

Scientific visualization has been used in the fields of medicine, fluid mechanics and so on [1]. In the field of oil-gas exploration, 3D visualization of seismic data could display the precise structure of underground geology, and help the geophysicists to discover the oil-gas resources more effective, since seismic exploration is an important method for the oil-gas exploration [2].

VTK is a visual open source software package, and QT is a cross platform GUI development platform which are widely used to develop software. Luo develop a software based on VTK and MFC to process the medical data [3]. Qian reduced computation time by automatically controlling the screen resolution [4]. Gao developed a 3D seismic data visualization system Sgy3D on Windows, using Open GL and VC++ 6.0 joint programming [5]. Jiang implement 3D modeling and visualization software using QT [6]. In order to develop a software which can run on both Windows and Linux, we design and develop a 3D seismic data visualization software based on QT and VTK.

2. Architecture of VTK

2.1. Hierarchical structure of VTK

VTK consists of two sub-systems, kernel/compiled layer (C++ Class Library) and interpretation layer supported script languages (Java, Tcl/Tk, VTK Interpreted Interface).

In the compiler layer, the data structures and algorithms associated with image processing are encapsulated as a C++ object, which are easily invoked by other classes and programs in VTK. After compiling, dynamic link libraries are formed. It binding to a scripting language to form an interface for interpreted languages, called the interpretation layer, Using the interpretation layer, you can quickly develop new features of VTK, greatly improving the efficiency of programming.

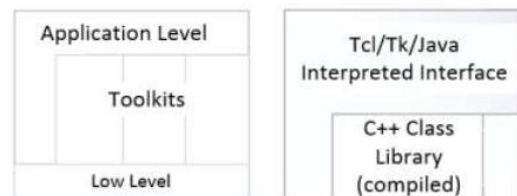


Figure 1. Hierarchical structure of VTK.

2.2. Framework of VTK

VTK uses data flow to transform the information we want to express into more intuitive graphical data. The data flow is illustrated in Figure 2,

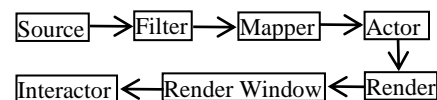


Figure 2. Data flow chart of VTK.

Where Source is the data source of VTK, used to read external data or create data. The Filter is a data processing mechanism with one or more inputs, but only one output. The purpose of filter is to process graphic image data in order to get what we expected.

Mapper uses these classes to map the application data after variety of filter processing to geometric data. It defines the interface between the raw data and the image data. Different Mapper can share the same input data, but render it in different ways. Actors: tells the actor the properties of the geometric data through the SetMapper() method, and then displays the results in the Window through Renderer.

RenderWindow is a real area on the computer screen, to display images. Interactor is the one that the user interacts with through the mouse.

3. Integrate QT with VTK

3.1. Establish the Integrated Develop Environment (IDE)

We choose VTK 5.0 + QT 4.8.4 + CMake 2.8 + Visual Studio 2010 to establish our IDE.

(1) Install QT

We download binary file and run it to install QT. After installation, we configure the PATH variable by adding QTDIR=D:\Qt\4.8.4

(2) Compile VTK from source code

We use CMake to manage the source code of VTK. Setting all the location of dependent header files and libraries, we can configure and generate the Visual Studio Solution (.sln) for the VTK.

(3) Configure the IDE

Copy the qvtk-widgetplugin.lib and qvtkwidgetplugin.dll to the directory of D:\Qt\4.8.4\plugins\designer, then restart QT.

3.2. Programming based on QT and VTK

Because VTK only provides 3D visualization libraries, developing software based on VTK directly is a console application. The console application lacks flexible and practical user interface. We must use other software packages, such as QT, MFC, FLTK, to develop a interactive 3D graphics processing system. Currently, the GUI frameworks supported finely by VTK are MFC and QT, where MFC is the classical GUI framework under the Windows platform, but the technology is relatively old and more complicated to use. Qt GUI is a cross platform framework based on the graphical user interface. We can quickly and easily create software with a friendly GUI. Mixed programming with VTK and Qt, can develop the cross platform GUI visualization software.

VTK and QT mixed programming refers to the develop a GUI software which can read ,process and render seismic data based on VTK and QT. In mixed programming, you need to embed the rendering window of VTK into the QT Widget. There are two ways to do that: (1) using QVTKWidget and (2) Copy the images byte by byte. Here we use the former, to integrate the QVTKWidget into Qt designer.

When using Qt to develop GUI, we uses Qt designer to design the GUI of 3D seismic data visualization software. After compiling, the system automatically generates the ui_xx.h file. We can modify the ui_xx.h file directly to operate on the GUI.

4. 3D Visualization of seismic data

4.1. Volume Rendering Algorithm

Volume rendering is a technique for obtaining two-dimensional projection images from discrete 3D data fields. The commonly used algorithms for volume rendering include ray casting, footprint, error shearing, 3D texture mapping and so on. The ray casting algorithm is used in this paper. It can be obtained from the rendering equation, and its imaging quality is very high.

Ray casting algorithm is a direct volume rendering algorithm based on image sequences. From each pixel of the image, along the fixed direction (usually along the line of sight) send a light beam across the whole image sequence, and in the process of image sampling and color information, based on the light absorption model of the color values are accumulated. Until the light across the whole image sequence, we finally obtained the color value is the color of the image rendering. The ray casting algorithm generally consists of four steps: project light, sampling, shading, and mixing. The basic flow of volume rendering using VTK is shown in figure 3.

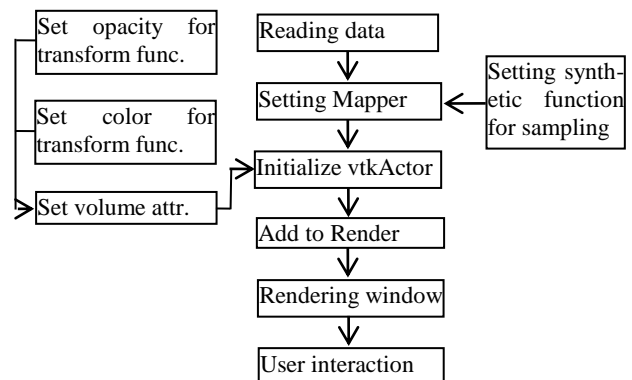


Figure 3. Flow chart of Volume Rendering using VTK.

In VTK, several important functions involving ray casting algorithms, opacity transfer functions, gradient opacity functions, color transfer functions, and illumination and shading are considered. The opacity transfer function (CompositeOpacity) is a piecewise linear scalar mapping function that can be used to set the final color by setting different opacity values.

```

vtkSmartPointer<vtkPiecewiseFunction> comOp =
  vtkSmartPointer<vtkPiecewiseFunction> ::New();
comOp->AddPoint(10, 0.00);
comOp->AddPoint(20, 0.10);
comOp->AddPoint(40, 0.30);
comOp->AddPoint(90, 0.40);
comOp->AddPoint(180, 0.50);
  
```

The gradient opacity function (GradientOpacity) maps the gradient modulus to an opaque multiplier, thereby enhancing the effect of the transition region.

Color transfer function (ColorTransferFunction). This article uses the RGB model to set the color by adding RGB breakpoint values.

```
color->AddRGBPoint(0,0,0,1);  
color->AddRGBPoint(127,1,1,1);  
color->AddRGBPoint(255,1,0,0);
```

4.2. Develop software GUI

We develop the software GUI using the Qt designer, and connect signals and slots. We implement the following functions:

- 1) the interface menu bar, drop-down menu bar, toolbar, status bar.
- 2) management of seismic data: horizontal data or longitudinal data can be sliced.
- 3) child window: color selection, slice coordinate control.
- 4) exit and help windows.

4.3. Display result

We develop the software on the Windows OS with the development environment of VS2010, VTK5.10 and QT4.8, and related environmental configuration. The 3D seismic data visualization software can display seismic data in a 3D window, and can make 3D interactive operation of the model including zooming, transverse cutting and longitudinal cutting. The main window of software is shown in Figure 4.

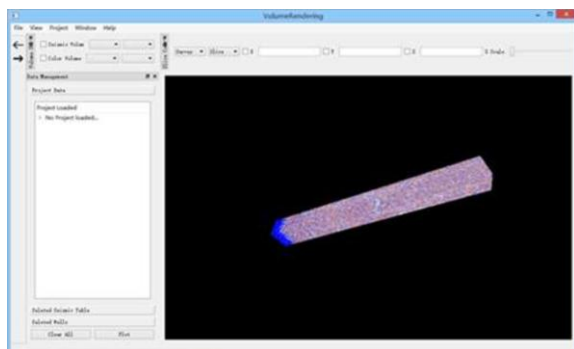


Figure 4. 3D seismic data visualization

5. Conclusion

This paper briefly introduces the architecture of VTK, the mixed programming of QT and VTK, and designs and develop a 3D visualization software for the seismic data. The software is tested for rendering a demo 3D seismic data successfully.

6. Acknowledgements

This work is supported in part by the National Science Foundation of China (No. 61273179, No. 61673006), and Science and Technology Research Project of Education Department of Hubei Province

of China (No. D20131206, No. B2016034, No. 20141304).

7. References

- [1] Kim, K., Carlis, J.V., Keefe, Daniel F., "Comparison techniques utilized in spatial 3D and 4D data visualizations: A survey and future directions", *Computer & Graphics*, In press, corrected proof, Available online 19 June 2017.
- [2] Chen, Z., Yang, H., Wang, J., et al., "Application of high-precision 3D seismic technology to shale gas exploration: A case study of the large Jiaoshiha shale gas field in the Sichuan Basin", *Natural Gas Industry B*, Vol. 3, No. 2, 2016, pp. 117-128.
- [3] Luo, H., Xu, Y., Chen, S., "The study and realization for 3D reconstruction of medical image based on VTK and MFC", *Progress in Biomedical Engineering*, 2010, Vol. 31, No.1, pp. 23-28,46.
- [4] Qian, S., Zhang, Y., "Key Technology of Seismic data 3D Visualization Display based on VolumeViz", *Computer System Application*, 2017, Vol. 25, No. 7, pp. 286-290.
- [5] Gao, J., Cui, R., Xu, C., "Study on 3D Seismic Data Visualization for Coal Mines Based on OpenGL", *Chinese Journal of Engineering Geophysics*, 2008, Vol. 5, No. 2, pp. 157-163.
- [6] Jiang, T., Chen, Z., Xu, X., "Study on 3D Visualization of Stratified Geological Objects Based on VTK and QT", *Science Technology and Engineering*, 2015, Vol. 15, No. 25, pp. 169-174.