Basic Research on Key Techniques Related to Urban 3D Pipe Network Modeling

DU Guoming  GONG Jianya  XIONG Hanjiang

1 Introduction

Urban pipe networks are composed of complicated underground spatial features, which include pipelines of water supply, drain, telecommunication, electric power, etc. Scientific methods must be used to plan underground pipe networks. Two-dimensional plane pipe network information systems have been applied successfully and widely. However, it is difficult to display 3D objects and surroundings by 2D technique. People pay more attention to the research on 3D technique, which makes people observe the objects vividly from different aspects. From 2D to 3D, the capacity of data becomes larger and spatial relationship changes accordingly. Because the implementation of 3D is difficult, the establishment of 3D data model is important. A new geometric modeling approach is presented in this paper.

2 Modeling principle of 3D pipe network

Because the pipeline is usually curved, a pipeline can not be simply treated as a cylinder, but can be treated as two parts: a cylinder and a joint between two cylinders.

2.1 Design the cylinder

Suppose there are arbitrary two points: \( P_1 (X_1, Y_1, Z_1) \), \( P_2 (X_2, Y_2, Z_2) \). Regard the two points as extreme points of pivot and employ grid modeling method to form 3D pipelines (supposing that the radii of the two sections of the pipeline are \( R \) ). Points \( (x_0, y_0), (x_1, y_1) \) are two adjacent points located at edge of section 1 which is a circle formed by point \( P_1 \) as center of circle and \( R \) as radius (see Fig. 1). Points \( (x_2, y_2), (x_3, y_3) \) in the section 2 which is a circle formed by point \( P_2 \) as...
center of circle and \( R \) as radius correspond to points \((x_0, y_0), (x_1, y_1)\) in section 1. The four points constitute a rectangle. When each circle covers \( n \) points, we can get \( n \) rectangles. The \( n \) rectangles constitute the surface of a prism. When the parameter \( n \) is small enough, the prism is regarded as a cylinder. The larger the number of \( n \) is, the better the effect is. But calculating capacity will increase when \( n \) becomes larger. When we design the system, \( n \) can be treated as a variant. It will increase while figures zoom in, and decrease while figures zoom out.

![Illustration of grid modeling](image)

Then coordinates of grids are calculated. The section of a pipeline is a circle (see Fig. 2). We suppose that the origin is located at the center of screen. Axis \( X \) is horizontal and directly rightward, axis \( Y \) is vertical and upward and axis \( Z \) is vertical and outward. We translate the origin to point \( P_1 \), then rotate axes system and keep axis \( Z \) in the same direction as vector \( P_1P_2 \), and axis \( X \) is vertical to original axis \( Y \). The angle between axis \( X \) and original axis \( X \) keeps acute angle. According to right hand theorem, axis \( Y \) can be determined. The section is displayed in Fig. 2. The coordinates are as follows:

\[
\begin{align*}
  x &= R \cos \alpha \\
  y &= R \sin \alpha \\
  z &= 0
\end{align*}
\]

where \( R \) represents the radius of circle; \( \alpha \) is angle between axis \( X \) and the radius.

The adjacent four points (just as displayed in Fig. 1) constitute a grid plane and the normal is calculated (see Eq. (2)). This is prerequisite to 3D modeling. The normal of each plane is vertical to grid plane and points outward. The formula is:

\[
\begin{align*}
  N_x &= R \cos(\alpha/2) \\
  N_y &= R \sin(\alpha/2) \\
  N_z &= 0
\end{align*}
\]

After the original coordinates are calculated, we must transform them into last coordinates.

When \( c_2 \neq \pm 1 \), it will be

\[
\begin{pmatrix}
  X \\
  Y \\
  Z
\end{pmatrix} = \begin{pmatrix}
  c_3 / \sqrt{1 - c_2^2} & c_2 / \sqrt{1 - c_2^2} & c_1 \\
  0 & \sqrt{1 - c_2^2} & c_2 \\
  -c_1 / \sqrt{1 - c_2^2} & -c_2c_3 / \sqrt{1 - c_2^2} & c_3
\end{pmatrix} \begin{pmatrix}
  x \\
  y \\
  z
\end{pmatrix}
\]

When \( c_2 = \pm 1 \), then

\[
\begin{pmatrix}
  X \\
  Y \\
  Z
\end{pmatrix} = \begin{pmatrix}
  1 & 0 & 0 \\
  0 & c & 0 \\
  0 & -c & 0
\end{pmatrix} \begin{pmatrix}
  \Delta x \\
  \Delta y \\
  \Delta z
\end{pmatrix}
\]

where \( x_0 = (x_1 - x_0)R \tan(\theta/2) \), \( y_0 = (y_1 - y_0)R \tan(\theta/2) \), and \( z_0 = (z_1 - z_0)R \tan(\theta/2) \).

2.2 Design the joint between two pipelines

It is simple to implement the linear interpolation. But the effect is not ideal. We employ another method. The formula is:

\[
\begin{align*}
  x &= 2R \cos^2 \omega \cos \varphi \\
  y &= 2R \cos \omega \sin \varphi \\
  z &= R \sin(2\omega)
\end{align*}
\]

where \( 0 \leq \omega \leq 2\pi \), \( \omega = \pi/n \), \( 0 \leq \varphi \leq \pi - \theta \) (\( \theta \) is the angle between the two pipelines). Fig. 3 shows the calculation of the coordinates of the interface. The normal is calculated by Eq. (6).

\[
\begin{align*}
  N'_x &= N_x' \sqrt{N'_x^2 + N'_y^2 + N'_z^2} \\
  N'_y &= N_x' \sqrt{N'_x^2 + N'_y^2 + N'_z^2} \\
  N'_z &= N_x' \sqrt{N'_x^2 + N'_y^2 + N'_z^2}
\end{align*}
\]
where
\[
\begin{align*}
N'_x &= (y_2 - y_0)(z_1 - z_0) - (y_1 - y_0)(z_2 - z_1) \\
N'_y &= (x_1 - x_0)(z_2 - z_0) - (x_2 - x_0)(z_1 - z_0) \\
N'_z &= (y_1 - y_0)(x_2 - x_0) - (y_2 - y_0)(x_1 - x_0)
\end{align*}
\]

Fig. 3 Calculating interface’s coordinates

Fig. 4 shows the calculation of the normal’s coordinates.

Now, we finish modeling the 3D pipelines. We establish the function of drawing figures: \texttt{DrawPipe3D(CDC \ast pDC, POINT3D \ast pCoor3D, int P, float R, int n)}$, where $pCoor3D$ represents the pointer to 3D coordinates, which answer for positions and directions of pipelines; \(P\) represents the number of the coordinates; \(R\) is the radius of the pipeline; \(n\) represents the units of grids, which is responsible for display effect. We can call the function directly to display the figures without rotation and translation of coordinates. Many graphics libraries such as OpenGL provide the function for displaying a cylinder for users. However, they do not provide the function for displaying a joint between two pipelines. The modeling method of 3D pipe network can effectively and quickly calculate the 3D coordinates of pipelines and their joints.

3 Implementation of 3D pipe network visualization and introduction of system functions

Modeling 3D pipe network is prerequisite to implementation of visualization. In this section, we introduce the implementing procedure of 3D pipe network visualization and the system function.

3.1 Implementing procedure of 3D pipe network visualization

The method introduced in this paper can run on the basis of OpenGL. In addition, it can run on other platform such as DirectX etc. The steps are described as follows.

1) Input data from Geostar, including object identifier-OID, radius, depth, coordinates of each pipeline and number of pipelines. Coordinate \(Z\) represents elevation.

2) Check the coordinates of each point and remove the redundant points. For example, when three points locate the same line, we can remove the middle point. Because radius, depth and coordinates of points are stored in attribute database and spatial database respectively. We can use special data structure to store them to improve the efficiency. (see Table 1)

3) Adjust the order of start and end points of pipelines and keep them in order. Combine the pipelines with the same attribute into one pipeline to decrease the middle points. (see Fig. 5)

4) Establish the model of 3D pipe network. Calculate the coordinates of pipelines’ surfaces and coordinates of the normals.

5) Generate display lists. The display lists are stored in advance in order to call afterward, which can speed up display of figures and optimize the

| Symbol | \(fn\) | Feature | \(ln\) | OID | \(R\) | \(Pn\) | Point3D |
|--------|-------|---------|-------|-----|------|------|---------|
| Explanation | Number of features | Features | Number of pipelines | Object identifier | Radius | Number of points | 3D coordinates |

Table 1 Data Structure of Pipelines

3) Adjust the order of start and end points of pipelines and keep them in order. Combine the pipelines with the same attribute into one pipeline to decrease the middle points. (see Fig. 5)

4) Establish the model of 3D pipe network. Calculate the coordinates of pipelines’ surfaces and coordinates of the normals.

5) Generate display lists. The display lists are stored in advance in order to call afterward, which can speed up display of figures and optimize the
system performance.
6) Clear buffers, including accumulation buffer, color buffer, stencil buffer, depth buffer.

![Fig. 5 Adjustment of the order of start and end points of pipelines and coalescing of pipelines](image)

7) Enhance display effect
OpenGL provides a series of library functions which can be used for establishing light model. The users can set up light model by modifying composition of ambient, diffuse, specular and position lighting and factor of lighting attenuation to simulate real light. The display of 3D figures is the effect of interaction between property of objects and light. Then we define material parameters to improve the simulative effect of 3D pipe network.

Fig. 5 Adjustment of the order of start and end points of pipelines and coalescing of pipelines

We can adopt antialiasing technique to eliminate the aliasing such as squared-round-corners. The function is gl-hint in OpenGL.

Furthermore, we can map texture into the 3D pipelines to improve visualization.

8) Display figures. Call lists to display figures. Fig. 6 is part of pipelines. The effect can be controlled by parameter $n$. When the parameter $n$ is small, the straight pipelines are prisms. When the parameter $n$ is large enough, the straight pipelines look like cylinders. But the calculating capacity will increase. The joint linking two pipelines looks smooth. The display effect of figures is good. (see Fig. 6)

![Fig. 6 3D pipe network's display](image)

3.2 Introduction of system functions
1) Various operations: zoom in, zoom out, translate, rotate and fly.

The system is provided with functions of window, center and view zoom, in general, it has the methods of coordinates translation or viewport translation. This paper adopts the latter, which uses function glTranslate to change the distance between object and viewer so that the figures can zoom. The method can decrease the calculating capacity and it is easy to design program with the method.

The function of rotation is glRotate. We can use mouse to rotate the pipelines and observe them from different aspects.

In order to observe the pipelines dynamically, we can imagine ourselves to fly along certain pipelines just like taking an airplane. That helps to observe or analyze the pipe network.

2) Bi-directional query. The system can get attribute information from the database by the object identifier-OID which is read from Geostar. When the users select some pipelines, a dialog box pops up to display attribute information. In the same way, when the users input the query condition, the pipelines which satisfy query condition will be displayed. (see Fig. 6)

3) Spatial analysis includes analysis of bursting pipelines, analysis of section and forecast of water consumption.

4 Efficient methods of speeding up display of figures
When data capacity is large, speeding up display of figures is important. Several methods we concluded are presented as follows:

1) Generate display lists. It is different from the sub-function. Because it is command of buffer instead of that of dynamic database, it can speed up the display of figures and optimize the system performance.

2) Create the special file which helps to input data and display figures. The file includes object identifier, number of pipelines, radius and 3D coordinates so that the system need not read data from
attribute and graphic databases. When the users input the data from Geostar, the file is created. Although it will increase the work capacity of input data, it may decrease the following work capacity.

3) When the amount of data is not very large (below the main memory), the data will be read into main memory one time.

4) Modify the parameter \( n \). When the figures zoom in, parameter \( n \) becomes larger. When the figures zoom out, parameter \( n \) becomes smaller. So it can satisfy simulative effect and speed up display of figures.

5 Conclusion

Modeling pipe network is prerequisite to establishing a pipe network information system. The method of modeling can influence speed and assignment of system resource, which will determine whether the users can be satisfied. Zoom, translation and spatial analysis is criterion in determining whether the system is good.

References

1 Qiao L, Fei G Z (2000) Program design of OpenGL. Beijing: Tsinghua University Publishing House. (in Chinese)
2 Tian Y Z, Zhang Z L, Yang X (1997) Exploration and information management of underground pipelines. Beijing: Publishing House of Surveying and Mapping. (in Chinese)