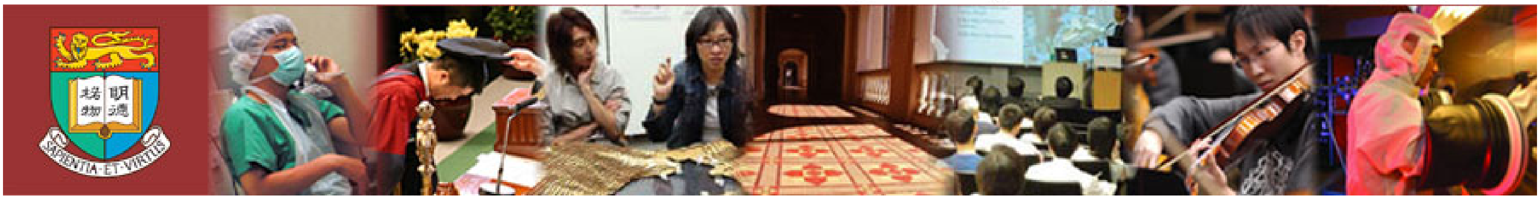

Title	VISJET-a computer ocean outfall modelling system
Author(s)	Cheung, SKB; Leung, DYL; Wang, W; Lee, JHW; Cheung, V
Citation	Computer Graphics International Proceedings, Geneva, Switzerland, 19-24 June 2000, p. 75-80
Issued Date	2000
URL	http://hdl.handle.net/10722/53605
Rights	Creative Commons: Attribution 3.0 Hong Kong License

VISJET - A Computer Ocean Outfall Modelling System

S.K.B. Cheung D.Y.L. Leung W. Wang
Department of Computer Science and Information Systems
The University of Hong Kong
{kbcheung, yldleung, wenping}@csis.hku.hk

J.H.W. Lee V. Cheung
Department of Civil Engineering
The University of Hong Kong

Abstract

Sewage and industrial effluents from coastal cities are often discharged into the adjacent sea after some land-based treatment. In modern design, the wastewater is often discharged in bouyant jet groups from risers mounted on a submarine outfall on the seabed to achieve rapid mixing of effluents with tidal flow. A mathematical model for bouyant jets in currents based on the Lagrangian models, called JETLAG, was developed. This paper presents a system, called VISJET, for visualizing the ocean sewage discharge system based on the JETLAG model. We will discuss the features of VISJET system and show how computer visualization can be used to help with the design of an ocean sewage discharge system.

1. Introduction

Sewage and industrial effluents from coastal cities are often discharged into the adjacent sea after some land-based treatment. In modern designs, the wastewater is often discharged in the form of rosette-shaped buoyant jet groups from risers mounted on a submarine outfall on the seabed. Rapid mixing of the effluent with tidal flows can be achieved by jet group design so that the water quality can be controlled to within acceptable levels. The hydraulics of submerged buoyant jets is intimately related to the design of effective wastewater disposal systems.

VISJET is a general predictive, PC-based, flow visualization tool to portray clearly the evolution and interaction of multiple buoyant jets discharged at different angles to ambient tidal current. It combines a general Lagrangian jet mixing model, JETLAG [1], with computer graphics techniques to trace the path and mixing characteristics of a group of arbitrarily inclined jets in three dimensional space. The model enables the determination of the degree of merging of interacting plumes. The ultimate objective is to develop an interactive tool for outfall design and environmental impact assessment.

A key component of JETLAG has been adopted for use for the environmental impact assessment of the Hong Kong Strategic Sewage Disposal Scheme, the Stage II Shanghai Combined Sewerage Project, and the post-operation monitoring of the Sydney Deep Water Ocean Outfall.

The main goal of VISJET is to provide:

- a friendly user interface for setting design parameters;
- informative graphics for quantitative analysis of merging plumes; and
- 3D interactive display and stereoscopic flow visualization.

The following terms will be used frequently later in this paper.

Jet: A continuous, steady discharge with an initial momentum.

Plume: A continuous, steady discharge with an initial density difference with the ambient (buoyancy).

Trajectory: The pathline tracing the location of the center of the jet.

For simplicity, we will use the first two terms – jet and plume – interchangeably hereafter.

In the following sections, we will put our emphasis on the visualization system and discuss how graphics techniques are used to assist engineers. In section 2, we will discuss the previous work in scientific visualization. In section 3, the system overview will be presented. The main features of VISJET will be presented in section 4. Transparency rendering, a problem encountered in VISJET, will be discussed in section 5. Finally, the conclusion for this project will be presented.

2. Previous Work

Scientific visualization has long been one of the most significant research topics in computer graphics. The basic idea of scientific visualization is to use computer-generated pictures to display information and perform data analysis. Computer-aided design, finite element analysis, simulation, and computational fluid dynamics analysis are just few examples to which visualization can be beneficial.

Recently, visualization on environmental data has become a hot topic. Xiao et al. [10] discussed the challenges in visualization of environmental data. Several systems have been developed for visualizing environmental data. A system for visualizing rainfall events has been developed in [9] and information on the mesoscale of rainstorms is obtained. Nations et al. [12] developed an interactive system of ocean circulation models.

The visualization system of water quality data developed by Forgang et al. [11] is similar to our system in several ways. Both systems are about environmental data and are used in the analysis of water pollution problems. However, the usage of their system is to compare simulated and measured water quality data, while our main focus is to assist the engineers to design a sewage discharge system.

3. System Overview

The purpose of VISJET is to provide a tool for civil and environmental engineers to visualize the effect of a sewage discharge system. This can help them design the actual sewage discharge system. The basic requirement of developing VISJET is to provide a visualization tool for JETLAG model. Figures 1 and 2 show the initial setup of the sewage discharge system. VISJET consists of two main components: JETLAG computational model and computer visualization system.

3.1. JETLAG

For environmental impact assessment and outfall design studies, it is desirable to take into account the effect of an ambient current on the initial mixing of bouyant wastewater discharges. The prediction of the concentration of a pollutant or passive scalar along the unknown jet trajectory of a buoyant effluent discharge is a complicated fluid mechanics problem which is not fully resolved. In particular, there are very few mathematical models that can treat satisfactorily a three-dimensional jet trajectory, such as a horizontal jet into a perpendicular cross-flowing tidal current – a common outfall design configuration. For impact assessment, post-season monitoring and risk-analysis, it is necessary to have a model that is capable of giving predications for an arbitrarily-inclined buoyant jet in a crossflow – covering the entire range of ambient current velocities and stratification conditions. JETLAG is a Lagrangian model which has given reasonably good predictions for this problem.

JETLAG [1] is a mathematical model for buoyant jets in

Figure 1. Ocean outfall with vertical diffusers: The volcano-like structures are the sewage outfall diffusers for discharging effluents.

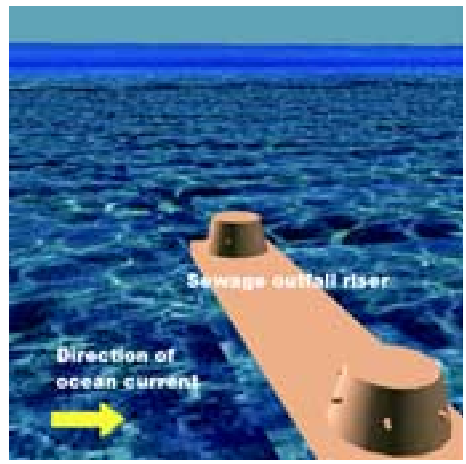

Figure 2. Sewage outfall diffuser.

currents, based on the Lagrangian model. It allows the user to find the trajectory of fluid coming out from a submarine jet, under the effect of ambient water currents. JETLAG is usually applied to the field of sewage discharge in order to predict the flow of pollutants. However, JETLAG model does not consider the interaction among multiple jets.

Input required by JETLAG are: the jet orientation, the local water depth, the vertical structure of ambient density and current speed and the jet discharge parameters. The model gives a 3D jet trajectory as output.

3.2. Visualization System

For the graphics part of VISJET, the following issues have to be addressed in its design:

- the system should enable one to visualize flow data with a sense of high realism;
- the users can interactively control the viewpoints and take a close-up picture; and
- the evolution of jets can be visualized using animation.

We will discuss the features of the visualization system in section 4.

4. Visualization System Features

In this section, we will discuss the features supported in VISJET and how these features can be used to help engineers. VISJET produces 3D visualization for the user to investigate the properties of jets with direct and intuitive visual aids.

4.1. Visualization

4.1.1. 3D Display

3D color graphics is used to display the spatial layouts of all jet trajectories output by JETLAG. When the user's virtual viewpoint or viewing direction is adjusted, the new 3D views can be displayed instantly to give the user real-time visual feedback. This feature also supports zoom-in and zoom-out to allow the user to have a close-up look at features of small scales.

Using surfaces puts the emphasis on the boundary of 3D geometric models. Using particles is good for animation and the randomness of fluids while displaying using vectors is suitable for showing the motion of discharge. We have chosen to model plumes using surfaces because the main focus of civil engineers is on regions confined by surface boundary.

4.1.2. Animation

The evolution of jets and other time-varying properties, such as velocity, can be displayed with special animation effects to enhance the understanding of the data displayed. The user can have a sense how the sewage discharges and evolves. Figures 3 and 4 are snapshots of animation showing the evolution of the jets.

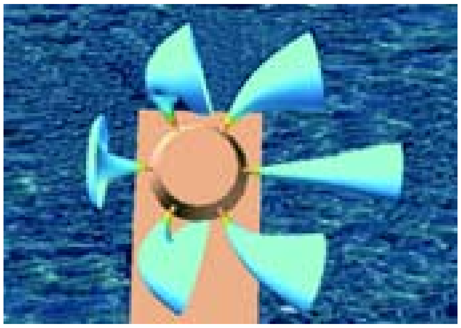

Figure 3. Rosette-shaped buoyant jets (overhead view): The effluents are being discharged in six jets.

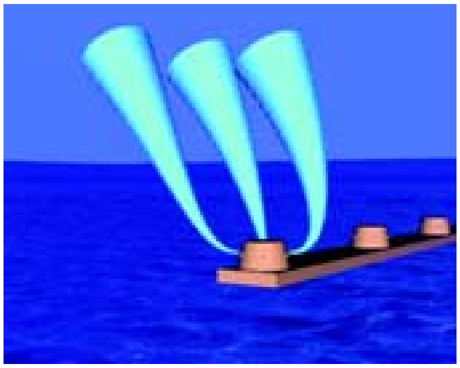

Figure 4. Rosette-shaped buoyant jets (perspective view).

4.1.3. Realism of ambience

External factors, such as direction of ambient water currents and reference objects (Figures 1 and 2) are displayed to provide a proper context for the data to be visualized. This gives the user a better sense of orientation and a better understanding of relationship between ambient factors and motion of jet trajectories.

4.1.4. Transparency and color coding

When multiple jets are displayed, they tend to appear in a cluster and occlude each other from certain viewpoints. This can be avoided by displaying all or most jets in a semi-transparent manner to alleviate mutual occlusion. However, rendering the jets with transparency requires a correct rendering order and this order is not easy to be found. In section 5, we will present an in-depth discussion on the issue and a hybrid usage of octree and binary space partitioning (BSP) tree in finding this order.

Color is assigned to the jet according to effluent concentration.

4.2. Data Analysis

4.2.1. Data interrogation

When the user wishes to know about data values defined at a point on a jet, such as velocity or concentration, this feature allows the user to locate the point of interest with a pointing device to interactively retrieve the data required. This kind of interaction is useful in data exploration and understanding.

4.2.2. Jet inspection by intersection

The user may use horizontal planes at different altitudes to intersect a particular jet and observe the resulting sections. This is helpful in understanding how each jet evolves when approaching the water surface. Figure 5 shows the result of using a horizontal cutting plane to show how jets merge.

4.2.3. Computing overlap among multiple jets

The situation of two or more jets overlapping with each other is interesting because it may signal that the pollution concentration in the overlapping region exceeds some threshold, or reaches a dangerous level.

One feature of VISJET computes and displays the area of and the concentration in an overlapping region of two plumes across a specified cutting plane. Global information about an overlapping region can be obtained using a series of parallel probing planes. This reveals the distributions of effluents at different places.

Figure 5. Merging of jet groups from adjacent risers.

5. Transparency Rendering

When multiple plumes are displayed, their trajectories tend to appear in a cluster and occlude each other from certain viewpoints. This can be avoided by highlighting one of the plumes and making others visually less conspicuous. An alternative is to display all or most trajectories in a semi-transparent fashion to alleviate mutual occlusion. One approach is to employ the techniques [3] in visible surface determination. Another approach is by modifying the underlying hardware to achieve the result. In the following, we will review these two approaches in more detail and finally, we will discuss our approach.

5.1. Visible Surface Determination

Visible surface determination is one of the fundamental tasks in computer graphics. The goal of visible surface determination is to determine which lines or surfaces of the objects are visible, given a viewpoint or viewing direction. One way is finding a visibility order such that a correct picture results if the objects are rendered in that order. Visibility ordering can be used in rendering with transparency.

Many algorithms have been developed to solve the problem of visibility determination. Recent works include the use of occlusion graphs [16] which resolves the visiblity problem based on the pairwise occlusion relation. This approach can find the ordering correctly but special work has to be done to handle the cycle problem (Figure 6a).

Let A, B and C be planar surfaces and the sequence of rendering based on pairwise occlusion relation is that C should be rendered before B, B before A and A before C. Then (A, B, C) form cycle of length 3.

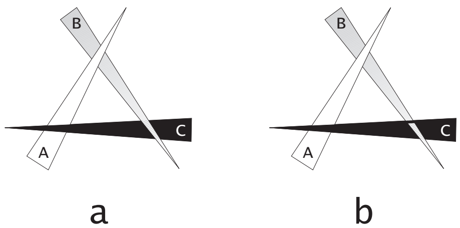

Figure 6. (a) A cycle exists. (b) By splitting polygon B, the cycle is broken.

The use of meshed polyhedra [7, 14] to obtain a total order generates a correct visiblity ordering of cells provided that the cells are convex and non-intersecting polyhedra. The algorithms can only detect the existence of cycles but cannot provide a solution if there is one.

5.2. Hardware Algorithm

Many algorithms for transparency rendering originate from those solving the problem of aliasing. Thus, most of the following are also solutions to antialiasing. Aliasing is the result of information loss due to sampling. To solve this, increasing number of samples per unit area is the only means and having more than one sample for each pixel is called supersampling. However, the drawback of supersampling is the requirement of large memory.

Several hardware algorithms have been proposed to tackle the problem. Implementation of A-buffer [4] algorithm can render transparency correctly if there are no interpenetrating semi-transparent surfaces, but this will require the objects to be sorted. The accumulation buffer [5] can maintain the quality produced by supersampling with less memory, but its multi-pass characteristics degrades the performance. The methods in [8, 13] also suffer from the problem of multi-pass rendering.

5.3. Our Approach

In our approach, we use both the octree and the BSP tree, which are usually used separately in tradition. First, the whole space is partitioned using an octree. This partition procedure will halt until there are fewer than a specified number of polygons in one cell. Then, within each leaf cell in the octree, a local BSP tree is used to partition the polygons (Figure 7). This is done in pre-processing. In the run-time, what we have to do is to use the BSP tree to find a correct visibility order.

One of the advantages of using this hybrid data structure is that both octree and BSP tree can provide a correct visibility order trivially. Another advantage is that there is no need to handle the problem of cycles because BSP tree breaks the cycle by splitting the conflicting polygons (Figure 6b). Also, this allows intersecting polygons to be rendered correctly. This hybrid approach can be used for rendering scenes with transparency. Moreover, both visiblity ordering and less restricted visible-surface determination can be handled in one single algorithm. The algorithm performs well in our application where the scene is static and only viewpoints and/or viewing directions can be changed. However, in this approach, both octree and BSP tree entail polygon splitting, and this normally increases the number of polygons.

Another advantage of our algorithm over algorithms using BSP tree solely is that the number of splitting can be made smaller when the visibility order can be resolved trivially.

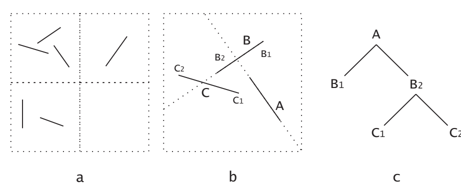

Figure 7. (a) Four cells of the octree. (b) Top view of scene with BSP tree before the recursion with polygon A as the first splitter. (c) The complete BSP tree.

6. Conclusion

We have presented a visualization system of an ocean sewage discharge system. This work enables the understanding of sewage discharge in 3D.

The focus of the paper is to show how computer graphics and visualization techniques can be used to help engineers in visualizing data and designing an actual sewage discharge system. Also, issues on transparency rendering and polygon intersections have been discussed.

This system has been found instrumental in the design and study of a pollutant discharge system. This tool is just the beginning of the whole story and will be enhanced to provide the users with more capabilities.

We have implemented the system on both Silicon Graphics workstations and PC. The development of a PC version makes VISJET more available to most industry and commercial users.

In the current version, VISJET includes the input interface to allow the user to adjust the parameters or configuration of the JETLAG model and generate the visual output interactively. This is essential when there are frequent changes of design for the sewage discharge system.

7. Acknowledgments

This project was partially supported by a HKU Teaching Development Grant.

References

[1] Joseph H. W. Lee, Valiant Cheung. Generalized Lagrangian Model for Buoyant Jets in Current. *ASCE Journal of Environmental Engineering*, pp. 1085–1106, November/December 1990.

[2] Hanan Samet. *Applications of spatial data structures: computer graphics, image processing, and GIS*, Addison-Wesley, 1990.

[3] James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes. *Computer graphics: Principles and Practice*, Addison-Wesley, 1996.

[4] Loren Carpenter. The A-buffer, an Antialiased Hidden Surface Method. *Computer Graphics, (SIGGRAPH '84 Proceedings)*, pp. 103–108, 1984.

[5] Paul E. Haeberli, Kurt Akeley. The Accumulation Buffer: Hardware Support for High-Quality Rendering. *Computer Graphics, (SIGGRAPH '90 Proceedings)*, pp. 309–318, 1990.

[6] Xianshu Kong, Hazel Everett, Godfried Toussaint. The Graham Scan Triangulates Simple Polygons. *Pattern Recognition Letters*, pp. 713–716, November 1990.

[7] Peter L. Williams. Visibility Ordering Meshed Polyhedra. *ACM Transactions on Graphics*, pp. 103–126, April 1992.

[8] Michael Kelley, Kirk Gould, Brent Pease, Stephanie Winner, Alex Yen. Hardware Accelerated Rendering of CSG and Transparency. *Computer Graphics, (SIGGRAPH '94 Proceedings)*, pp. 177–184, 1994.

[9] Lloyd A. Treinish. Severe Rainfall Events in Northwestern Peru: Visualization of Scattered Meterological Data. *IEEE Visualization '94*, pp. 350–354, 1994.

[10] Yingcai Xiao, John P. Ziebarth, Chuck Woodbury, Eric Bayer, Bruce Rundell, Jeroen van der Zijp. The Challenges of Visualizing And Modeling Environmental Data. *IEEE Visualization '96*, pp. 413–416, 1996.

[11] Adam B. Forgang, Bernd Hamann. Visualization of Water Quality Data for the Chesapeake Bay. *IEEE Visualization '96*, pp. 417–420, 1996.

[12] Scott Nations, Robert Moorhead, Kelly Gaither, Steve Aukstakalnis, Rhonda Vickery, Warren C. Couvillion Jr., Daniel N. Fox, Peter Flynn, Alan Wallcraft, Patrick Hogan, Ole Martin Smedstad. Interactive Visualization Of Ocean Circulation Models. *IEEE Visualization '96*, pp. 429–432, 1996.

[13] Stephanie Winner, Michael Kelley, Kirk Gould, Brent Pease, Bill Rivard, Alex Yen. Hardware Accelerated Rendering of Antialiasing Using A Modified A-buffer Algorithm. *Computer Graphics, (SIGGRAPH '97 Proceedings)*, pp. 307–316, 1997.

[14] Cláudio T. Silva, Josehph S. B. Mitchell, Peter L. Williams. An Exact Interactive Time Visibility Ordering Algorithm for Polyhedral Cell Complexes. *IEEE Symposium on Volume Visualization '98*, pp. 87–94, 1998.

[15] A. Clematis, M. De Martino. Using GIS to Visualise Environmental Information — A Case Study Based on Digital Terrain Models. *IEEE Computer Graphics International '98*, pp. 123–127, 1998.

[16] John Snyder, Jed Lengyel. Visibility Sorting and Compositing without Splitting for Image Layer Decomposition. *Computer Graphics, (SIGGRAPH '98 Proceedings)*, pp. 219–230, 1998.