

APPLICATION OF VISUALLISP PROGRAMMING LANGUAGE TO 3D SLUICE MODELING

Nguyen Thi Lan Truc¹ and Po-Han Chen²

¹Master of Engineering Student, School of Civil and Environmental Engineering,
Nanyang Technological University, Singapore 639798

²Assistant Professor, School of Civil and Environmental Engineering,
Nanyang Technological University, Singapore 639798

Abstract

Nowadays, it is convenient to use 3D modeling tools for general planning before construction. Normally, a 3D model is built with 3D CAD such as 3D Studio Max, Maya, etc. or simply with AutoCAD. All these software packages are effective in building 3D models but difficult to use, because many provided functions and tools require prior knowledge to build both 2D and 3D designs. Moreover, the traditional method of building 3D models is most time-consuming as experienced operators and manual input are required. Therefore, how to minimize the building time of 3D models and provide easy-to-use functions for users who are not familiar with 3D modeling becomes important.

In this paper, the VisualLISP programming language is used to create a convenient tool for efficient generation of 3D components for the AutoCAD environment. This tool will be demonstrated with the generation of a 3D sluice, an artificial passage for water fitted with a valve or gate to stop or regulate water flow. With the tool, users only need to enter the parameters of a sluice in the edit box and the 3D model will be automatically generated in a few seconds. By changing parameters in the edit box and pressing the “OK” button, a new 3D sluice model will be generated in a short while.

Keywords: AutoLISP, 3D sluice model, automatic generation

1 Introduction

It is convenient nowadays to use 3D modeling tools for general planning before construction. Most of the 3D modeling software packages are effective in building 3D models but sometimes not so easy to use, as prior knowledge of how to build 2D and/or 3D designs might be required. For instance, it takes around two days to build a simple building and several months to build a complex one. In this regard, how to minimize the building time of 3D models and provide user-friendly functions becomes crucial.

To meet this need, an application was developed as a tool that realizes automatic generation of 3D models. This tool was initially developed to generate 3D sluices, artificial passageways for water fitted with a valve or gate to stop or regulate water flow. With this tool, users no longer have to build 3D models step by step. The input of several parameters into the edit boxes followed by pressing the “OK” button will trigger the automatic generation of 3D models, which takes only a few seconds.

2 Automatic generation of 3D sluice model

2.1 Scope and contribution

The traditional way of simulating 3D models requires lots of manual input and, hence, takes a lot of time. The ordinary steps are as follows:

- Manually create the 3D components of a sluice one by one, based on its 2D blueprint. Some common components could be imported from the software library.
- Merge all components into a complete sluice.
- Create the landscape around the sluice.

Because a sluice is a specialized hydraulic facility, there is no library for sluice components and no way to save the time needed for its creation. Thus, both 3D modeling experience and sluice knowledge are required to create a sluice.

The new method can meet the two requirements by saving creation time without having 3D modeling knowledge. Instead of modeling every component, the user just needs to enter some parameters and the 3D model will be automatically generated. Any modification on components can be carried out by modifying the parameters in the edit boxes.

Nowadays, due to the popularity of the AutoCAD environment to most users, the VisualLISP programming language, which is included in AutoCAD, is chosen to implement the application. The application enables users to create 3D sluice models with the least manual input.

2.2 Overall application flowchart

Users can start this application by entering the required values. They are dimensions and options of sluice components as well as the insertion point of the sluice model in the 3D space. The input values will be processed to automatically generate each sluice component and subsequently merge all of them together. The overall flowchart is depicted in Figure 1.

In detail, the dimensions and options are used to create each sluice components and the insertion point is used as a base point to connect all related components. If there are any pre-

defined components in the library, they can be inserted directly to the sluice model. Finally, the complete model, which is in the form of wireframe, is shaded in color.

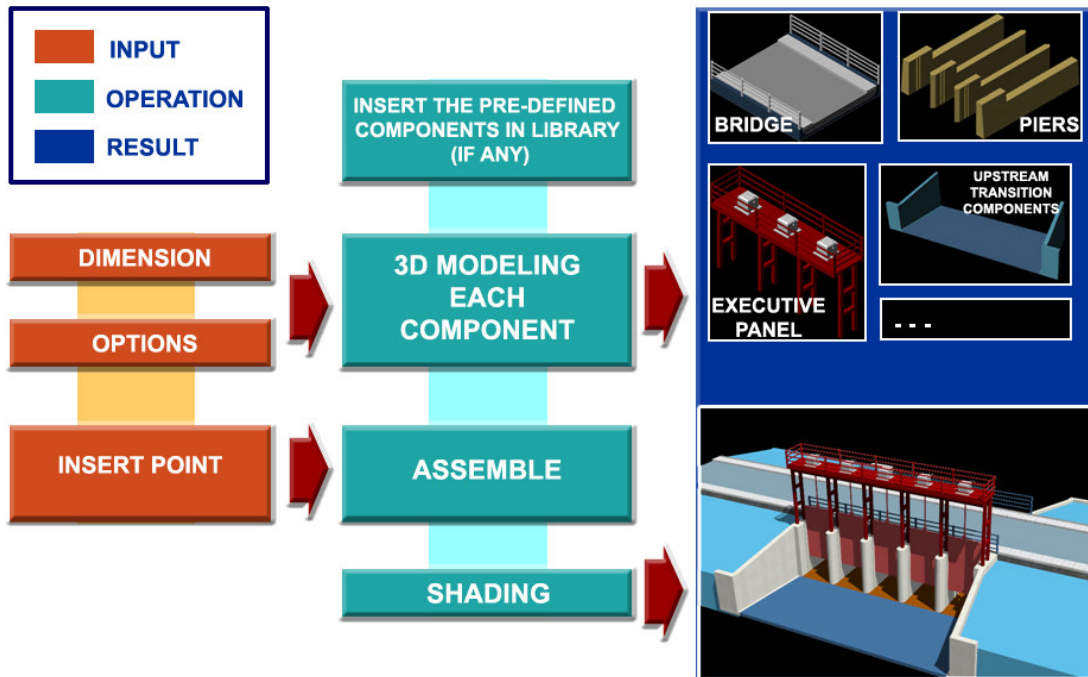


Figure 1: Overall application flowchart

2.3 Implementation steps

The steps of creating 3D models are similar to the ordinary steps mentioned in Section 2.1. The different is that the generation of 3D models is automatically done by VisualLISP instead of manual work.

In brief, the implementation steps of the application are as below:

- Distinguish different sluice components which have relationship with some functional blocks.
- For each block, identify important parameters that characterize each block's component as well as the whole block.
- Use the AutoLISP programming language together with AutoCAD commands to create functions which can build individual 3D components and assemble all components to form a 3D model (Figure 2).

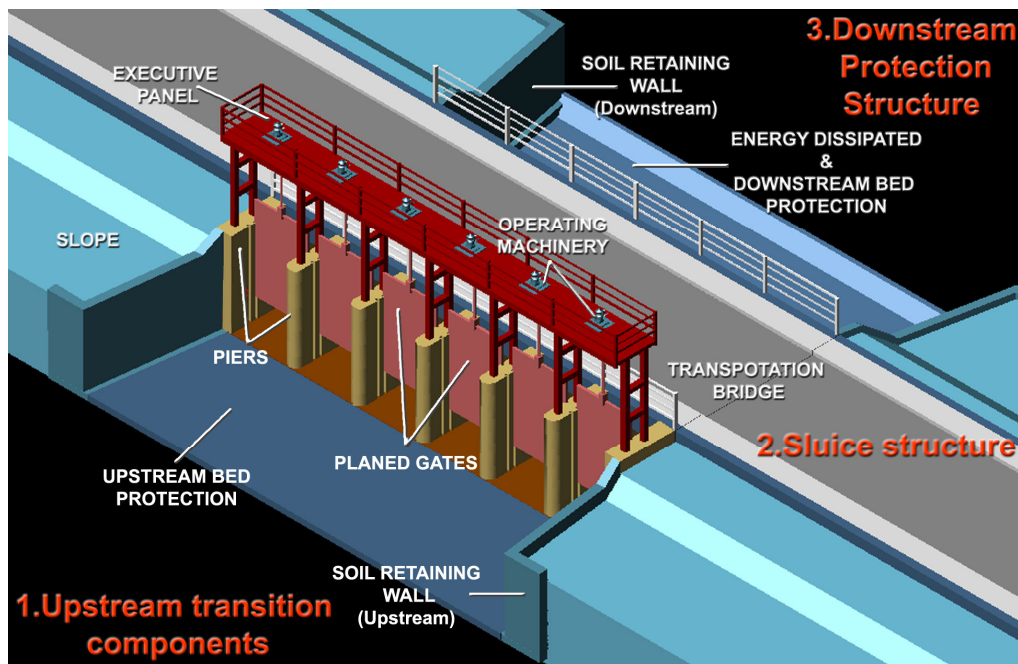


Figure 2: Illustration of 3D sluice model

- Using VisualLISP Dialog Control Language (DCL) to create the graphical user interface (GUI) which includes a dialog box for each functional block. The dialog box comprises edit boxes to input dimensions, radio buttons to select component's options and figures to illustrate 3D component images as well as to denote specific parameters that appear in dialog edit boxes.

Distinguish sluice components for functional blocks

According to books on hydraulic works, a sluice could be divided into 3 main blocks: Downstream Protection Structure, Upstream Transition Components and Sluice Structure.

Each block is constructed with many components as in Figure 3. By distinguishing different components carefully and naming each of them clearly, the 3D modeling could be constructed in detail and with accuracy.

Identify important parameters that characterize each block

The shape of each component relies on its dimensions. Therefore, the dimensions will be used as the input values to characterize the component.

Basically, to shape a component, it is not required to input all the dimensions, as some of them could be interpolated, calculated or obtained from other components. Thus, it is important to recognize which parameter is necessary among various candidates. The less the number of parameters, the more user-friendly the application. Figure 4 represents the main parameters to characterize the upstream retaining wall of sluice.

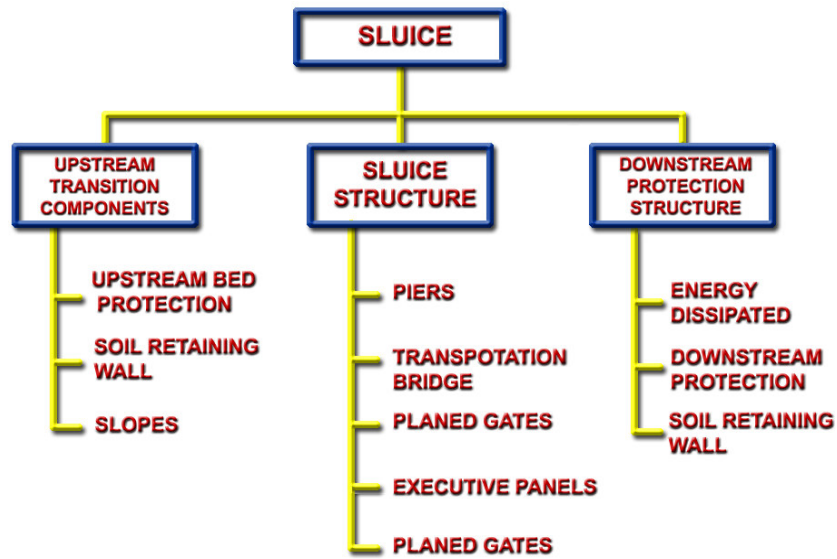


Figure 3: Sluice components

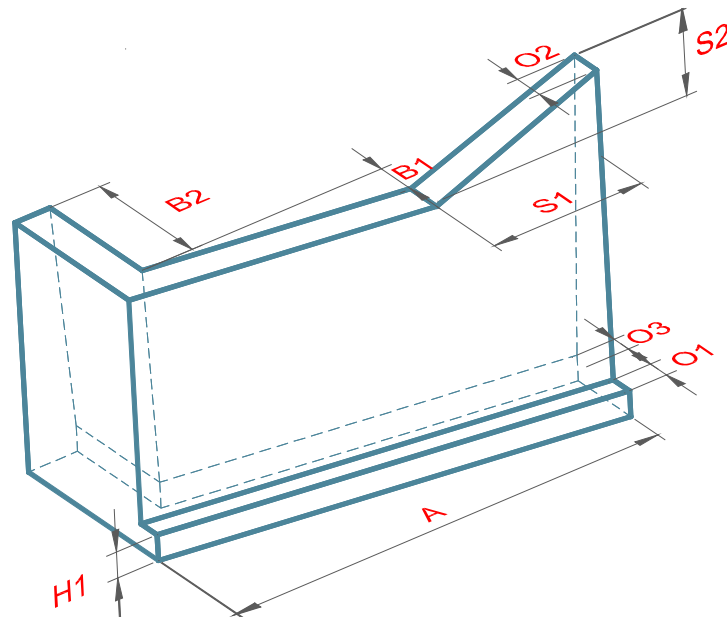


Figure 4: Parameters characterizing the upstream retaining wall of sluice

In the above figure, some of the parameters could be re-used or used to calculate the dimension of others components which are related to retaining wall.

The application programming functions

In this application, there are various functions written using AutoLISP commands, which are to automatically generate components, merge components, modify and shade.

For functions that are used to automatically generate 3D components, the modeling procedure basically follows the manual way of using AutoCAD commands.

In manual way, operators are required to type AutoCAD commands into the command window and enter correct parameters in the command line using keyboard. In this application, these above works are automatically done by AutoLISP.

To implement, the AutoLISP commands are edited in a text editor called VisualLISP for AutoCAD. VisualLISP supports the command “command” with which any AutoCAD command could be called and implemented accordingly. The parameters of each command are calculated first by AutoLISP operators and functions and then assigned to the corresponding command.

Let’s take a look at how to construct a function with which a soil retaining wall could be generated. The production is followed by the following steps and illustrated in Figure 5.

- The 2D boundary of wall’s section and the extruded path are generated by using AutoCAD “pline” command to connect all related points together. The coordinate of each point is calculated from input data.
- Using AutoCAD “extrude” command to generate the basic block of soil retaining wall by extruding the 2D boundary along the path.
- Trim any unnecessary parts. The size of the trimmed part is determined by calculation from the existing value.
- Using AutoCAD “mirror” command to get the opposite wall.

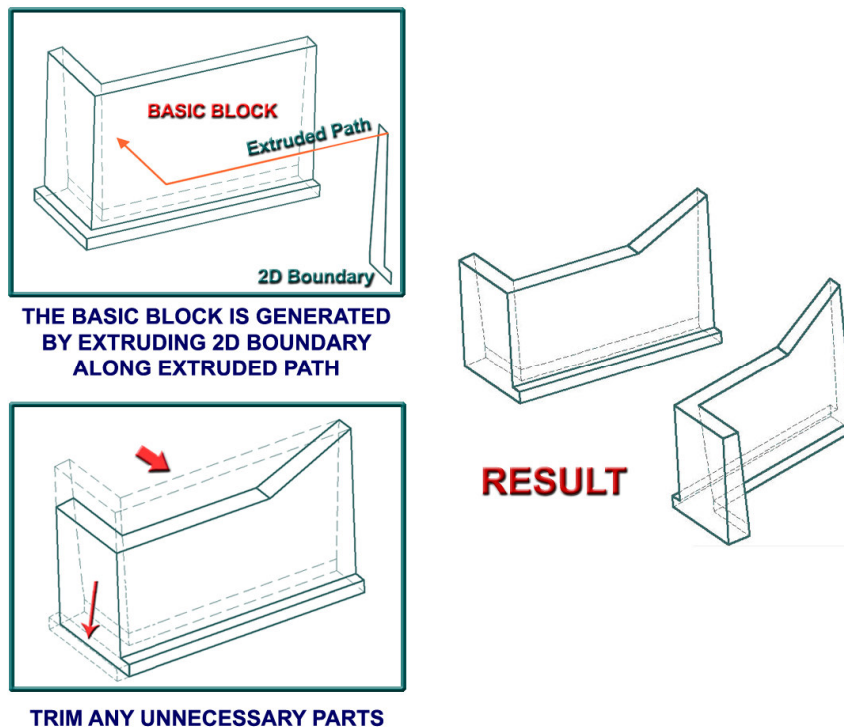


Figure 5: Illustration of the steps to generate soil retaining wall

In the above steps, the ordinary AutoCAD commands are implemented by using AutoLISP “command” command.

For instance, to extrude the 2D boundary along the path, the AutoLISP command should be written as

```
(command "extrude" TC1 "" "path" TC2 )
```

where:

“command” is AutoLISP command which is used to call an AutoCAD command.

“extrude” is an AutoCAD command to extrude a 2D boundary to a specific height or along an existing path.

TC1 is the name of 2D boundary and also is the 1st parameter of extrude command.

“path” is the option of extrude command to extrude the 2D boundary along a path not to a specific height.

TC2 is the name of extruded path and also the 2nd parameter of extrude command.

Similarly, the other components are using the same procedure to generate the 3D models automatically.

3 Graphical User Interface (GUI) of application

The way to create GUI (dialog boxes or widget) using VisualLISP is different from using other visual programming languages, as VisualLISP does not support the GUI wizards.

In VisualLISP, users are provided with Dialog Control Language (DCL), a build in GUI mini language for creating simple graphical dialogs within AutoCAD. DCL allows users to interact with the AutoCAD environment through basic widgets called *tile* such as edit boxes, buttons, checkboxes, radio buttons and list boxes.

In a DCL file, to construct a complete dialog, we have to list the entire tiles that will appear in the dialog in a reverse order. Hence, all tiles must follow the tree structure. That means the parent widget have to contain the children widgets successively. Figure 6 shows how a dialog is constructed using DCL.

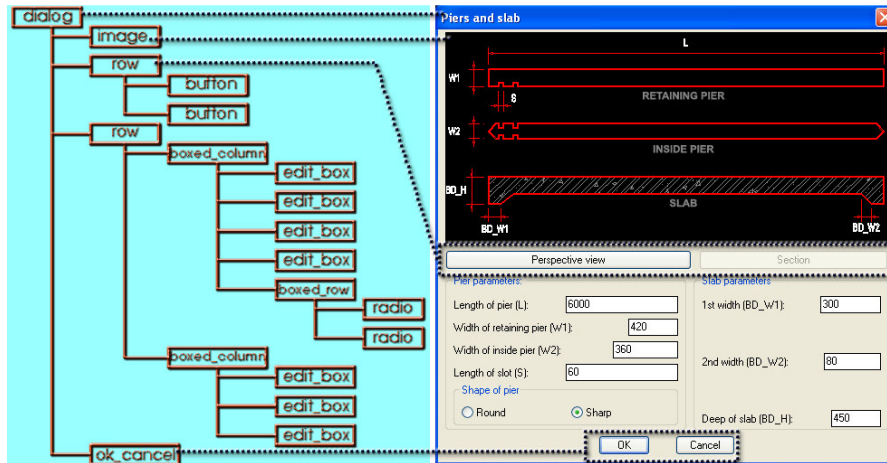


Figure6: A dialog box created using Dialog Control Language

4 Application facilities and result

Application facilities

This application provides the possibility of quick and automatic generation of 3D sluice models. Therefore, users who do not know much about 3D modeling could still create 3D models without any problem.

To achieve this, the application should provide user-friendly and dynamic GUIs for easy operation. Namely, users do not need to worry about the names of the component parts appearing in edit box labels but just follow the depiction of parameters on the dialog figure. Moreover, users are able to see the illustrated components both in the 3D perspective view and the 2D section view or plan view (Figure 7).

Because the 3D sluice model is generated in the AutoCAD environment, users could use AutoCAD utilities to improve the 3D model such as adding or modifying entities.

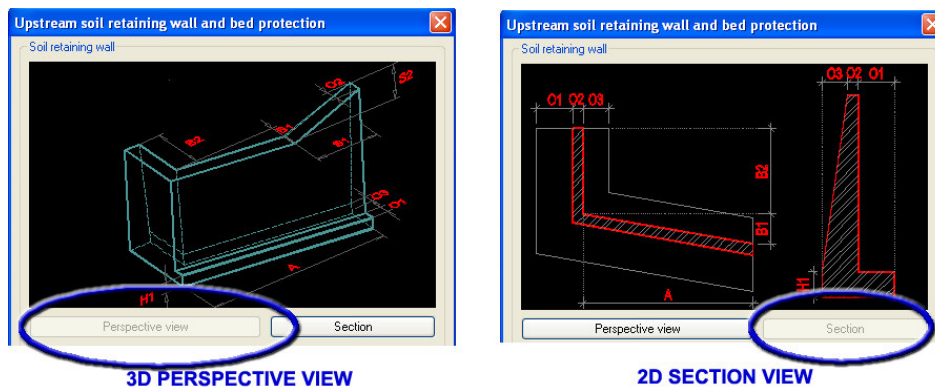
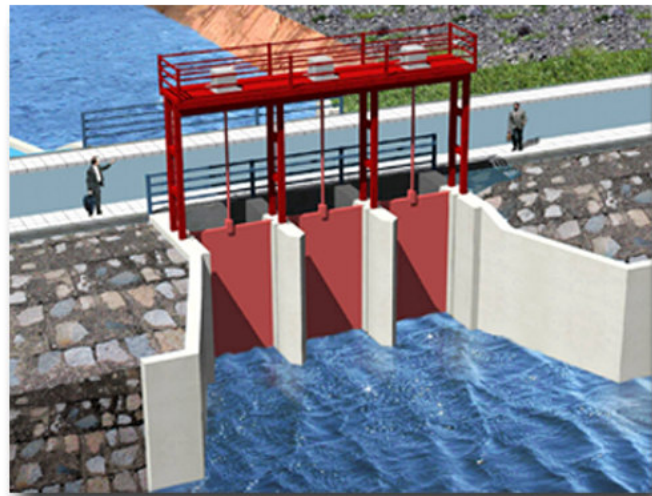
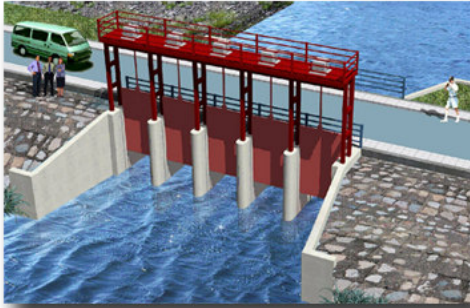


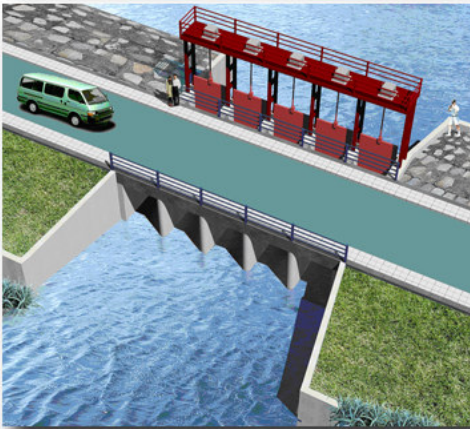
Figure 7: User-friendly and dynamic GUIs for easy operation

Application result

The application result is the generated 3D sluice model whose shape is determined by user input data. Figure 8 represents several rendered products of the 3D sluice model generated by this application.



**THE RENDERED IMAGES OF 3-GATES SLUICE
WITH SHARP PIER'S HEAD.
UPSTREAM VIEW**



**THE RENDERED IMAGES OF 5-GATES SLUICE
WITH ROUND PIER'S HEAD
UPSTREAM AND DOWNSTREAM VIEW**

Figure 8: Completed 3D sluice model

5 Conclusion

In this paper, the VisualLISP programming language is used to develop an application which could automatically generate 3D sluice models. With this application, users could produce their own 3D models in a short while without any extra manual input. In Figure 8, the generation time for the 5-gate sluice by a novice is around 10 minutes. Based on this application, more sophisticated 3D modeling using VisualLISP could be achieved by integrating additional features.