Masaryk University

Faculty of Informatics

Maya2CellVIEW: 3D Package Integrated Tool for Creating Large and Complex Molecular Scenes

Master's Thesis

David Kouřil

Brno, Fall 2016

MASARYK UNIVERSITY

FACULTY OF INFORMATICS

Maya2CellVIEW: 3D Package Integrated Tool for Creating Large and Complex Molecular Scenes

MASTER'S THESIS

David Kouřil

Brno, Fall 2016

This is where a copy of the official signed thesis assignment and a copy of the Statement of an Author is located in the printed version of the document.

Declaration

Hereby I declare that this paper is my original authorial work, which I have worked out on my own. All sources, references, and literature used or excerpted during elaboration of this work are properly cited and listed in complete reference to the due source.

David Kouřil

Advisor: Ivan Viola

Acknowledgement

This is the acknowledgement for my thesis, which can span multiple paragraphs.

Abstract

Scientific illustrators communicate the cutting edge of research through their illustrations. There are numerous software tools that assist them with this job. The aim of this thesis is to push abilities of illustrators working on a large scale molecular scenes. This is done by connecting two software packages - Maya and cellVIEW - combining the rendering possibilities of cellVIEW and modeling tools of Maya which results in more effective and efficient workflow.

Keywords

keyword1, keyword2, . . .

Contents

List of Tables

List of Figures

1 Introduction

In this age, scientists come to new findings everyday. Unfortunately, not all of these are ever shown to the general public. It could be because of several reasons. One of them might be that the new facts are not easily visualizeable. This can also mean that the things that they found out are a piece of a bigger picture. This is the job of scientific illustrator. These artists are usually experts in their own scientific domain and what they do, is they try to visualize the science in their domain with easily understandable images, animation and other media. There are many tools for this job. Few years back illustrations have been done by hand, drawings and paintings. Then came the age of computer graphics. As graphics tools have become more and more common and sophisticated in a field of computer generated movies (reference to Pixar?). Tools have become more accessible, easier to use and so scientific illustrators adapted and tried to use these tools for their benefit. Tools like Maya, Cinema4D, or XXX are nowadays the industry standards when it comes to authoring 3D content. Inside these programs there usually exist a software called renderer. With this two part - modeler and renderer - the user is able to create images or animations (movies) from the 3d data that he creates in the software. Even with just this, the modern computer graphics brings us the possibility to create amazing imagery and enables illustrators to show ordinary people fenomena from all kinds of science disciplines. But this workflow isn't very effective. One of the obstacles in the work of illustrators is that the software they are using is made for somebody else. Games and movies industry are the leading industries that push computer graphics tool creators forwards and provides most of the revenue for them. This means that these tools, no matter how much they try to be versatile, are being skewed towards the use cases of movies and games industries. That means that people that want to create scientific content might struggle to use the tools sometimes. For this, there are other tools that help them to make the pipeline more effective. This doesn't mean that these tools completely replace Maya or Cinema4D. The domain specific tools are more commonly implemented in a form of a plugin that is developed and loaded into the concrete software packages. In the end - the features that 3D authoring package provides for modeling 3D scenes are hugely beneficial to the creators. Another things is - people that used these software are used to that and switching from one to another is not a matter of a short time. That's one of the things that we need to take into account. The pipeline of each artist might be a little different and rather than suggesting a solution that completely replaces their pipeline, it's better to make a tools that doesn't break their workflow but rather the artist is able to incorporate the tool, benefit from it but doesn't have to change the way they work a lot. There also exist programs that are completely separated from any of the mentioned authoring software. There is an active research in the domain of visualization, with many conferences every year and thousands (?) of research papers in this domain. Usually, as a biproduct, these paper generate software that showcases the proposed visualization technique or pipeline. Some of these developed into full featured visualization packages and thus provide a way how to illustrate something in a new way. While these programs might hugely benefit scientific illustrators, it's not always the case that they get used. This might be because of several reasons one of which is that the illustrators are simply used to a certain pipeline and incorporating a new software into this pipeline doesn't seems very beneficial to them. Another problem is that because these programs are developed for a certain use cases (showcasing the point of the paper) they might not be easily appliable to more than one purpose.

In this thesis, we made an effort to solve these problems. Our domain of choice is molecular visualization. In this domain we try to visualize living (?) organisms on the smallest possible scale. With this approach, the computer memory and performance requirements are very challenging even for todays available hardware. Still, people have been able to visualize for example model of HIV in blood plasma in it's full detail down do individual atoms of each protein. This however, is achieved by using a very customized rendering method and a custom data format. In 3D graphics, data are usually represented as meshes consisting of triangles (which in turn are made of vertices). If we wanted to represent each atom as a sphere, we would need at least <number> of vertices for one atom and that's only for the roughest level-of-detail. With such representation, a one whole protein could use up to X bytes of memory which already takes up most of the video cards' memory. Thus this representation is not suitable for this task.

Instead, various other techniques have been developed for rendering on molecular data. That being said, these techniques are not usually taken into account when 3D authoring software is developed. As was mentioned, the primary users of these software are video games and movies industry, where mesh representation is the one used. That implies that people that use these programs don't have access to the cutting edge technology of rendering, visualization and illustration of molecular data and there is room for us to improve this situation.

In the next chapter, we will describe the state of the art tools that are used for molecular visualization today and we'll mainly focus on showing the gaps in intercompability of the available programs. Then, in chapter 3, we will propose a method of how the problem could be solved.

2 Introduction - old

2.1 Biological visualization and illustration

Here I should somehow present the field (but I don't know what field exactly - biological visualization, molecular visualization?)

2.2 Motivation

In this thesis we focused on providing better tools for scientific illustrators. The idea was to bridge software that they are used to (represented by Maya) and software that brings modern rendering abilities to the domain of molecular rendering (represented by CellVIEW). Emphasize the difference between offline and realtime rendering.

3 State of the art

In this chapter we will look at the current tools for authoring of scientific illustrations. Illustrators have gone a long way from doing these works by hand and nowadays there exist a variety of programs that help them do their job communicating the scientific findings. Barely any of these tools is perfect. Some of the tools are very closed and tied to it's creator. The ideal case would be if any new user could approach any of these tools and with a little training be able to use it. This is not the case, unfortunately. From the ones that are fairly simple from the user experience stand point is Molecular Flipbook[ref]. It provides user the ability to create simple molecular models and animate them with a keyframe-like animation workflow.

An important part of any scientific illustration are the data the one is trying to illustrate. In the case of molecular models, there is cellPACK which helps us with this. cellPACK is a software that assembles large molecular scenes from a description (a recipe) of how this scene should look, what it should contain and in what quantity.

3.1 Maya

Autodesk Maya is not a tool for molecular visualization per-se.

3.2 cellVIEW

4 OLD - Overview of the two programs that I'm going to be bridging - TODO better title

Here I will talk about in which context are these two 'systems' used and what is each for. The point of this project was to make a bridge between two programs. In this chapter we will look at both of them, see what they bring to the table when used separately, what they lack and how can be benefit from connecting them in this way.

4.1 CellVIEW

Briefly describe what CellVIEW is about. Give links to the papers that are describing CellVIEW and those that are using it.

4.1.1 Unity

I don't know if this is needed. I could just go little bit into detail of why is Unity good (which is funny now that we are probably going away from it)

4.2 Maya

Point to examples of animations that are produced with Maya. Maya is a 3D content authoring software package developed by Autodesk. It has been introduced in 1998 and since than it has grown into de-facto industry standard in many fields that use computer graphics (most notably movies and computer games industry). Because of it's versatility, Maya has also been used for creating scientific illustrations. Although Maya is a very robust piece of software that gives it's users ability to create various types of 3D scenes, it isn't always the best tool for the job. There exist other software packages that might be more specialized and better suited for a task at hand.

4.2.1 Extensibility

Luckily, Maya comes with an enormously useful ability to be extended with so called plugins. Autodesk provides an API that programmers can use to extend Maya's functionality. Several plugins have been developed specifically for the field of molecular visualization and illustration. As an example, we can name Molecular Maya which brings users the ability to import molecular data from online Protein Data Bank. The plugin then allows the user to choose how this data should be represented in a 3D scene and Maya then takes care of rendering such structure.

5 Method

I have introduced the field. I have sketched the problem that we are trying to solve (improving the workflow of animators/illustrators). Now it's time to describe the suggested solution. We talked about improving the workflow of scientific illustrator that uses Maya as an authoring software. The way that we are going to accomplish this is by taking advantage of extensibility of both Maya and Unity.

6 Implementation

6.1 WinAPI

6.2 Maya side

Topics - Architecture of the plugin, what parts/classes of API are used, memory layout, what I tried and didn't work.

6.3 Unity side

Topics - Architecture of the plugin, very generally about plugins (it's just a basic C++ dll plugin), interface between C++ and C#

7 Demonstration

In this chapter we present two use cases of how the user can approach illustration with this new proposed system.

8 Discussion, future work

What optimizations I could implement, how it could be used and what would be needed to implemented for that.

A An appendix

Here you can insert the appendices of your thesis.