Realistic Physically-Based Rain Simulation

Carlin Yuen

Advisor: Dr. Norman Badler, Joseph Kider

University of Pennsylvania

ABSTRACT

For my senior design project, we would like to propose the use of a physically based particle simulation which addresses more accurate creation of a mixture of weather-related effects in virtual environments, which will hopefully lead to the correct visualization and rendering of the photometric properties of these effects. I would like to approach this topic of study because current virtual environments lack appropriate computational methods to realistically simulate weather and local atmospheric effects due to its complex nature.

The project entails the creation of a computer software system for simulation and visualization of weather-related effects in virtual environments, specifically focusing on rain. In addition to modeling the environment and scenarios, we would need to effectively simulate realistic weather conditions to produce visually accurate illumination. Simulation automates the problematic tasks of hand fashioning weather and its effects on terrain yielding a broad assortment of appearance controls. Thus, the simulation designer merely has to adjust model parameters rather than recreate an entire scene, and consequently speed up the workflow of producing virtual environments with realistic weather effects.

Project Blog: http://www.carlinyuen.com/blog

1. INTRODUCTION

Computer graphics today has advanced to a point where realism is becoming a standard in many instances of animation and virtual reality today. In particular, scenes that involve weather effects in any manner—rain, snow, hail, fog, etc.—are significantly enhanced when the effect is recreated believably. Real-time applications of such natural phenomena in the past few years were limited due to hardware constraints that could not process the effects with satisfactory frame rates; however, recent developments in hardware have shown enormous potential for implementations that would incorporate both realism and speed[RJG06]. The current lackluster in realistic weather simulation is no longer hindered by graphics hardware, but rather, current virtual environments lack appropriate computational methods to realistically simulate weather and local atmospheric effects due to its complex nature. This is only exacerbated by the fact that realistic weather is often overlooked as a critical component to enhancing realism in video games and other such virtual reality implementations[Bar08].

The simulation of rain has traditionally fallen into two categories: static textures and particle systems (often used in video games for speed, while sacrificing realism), and physically-based methods (which incur a high computational cost)[RJG06]. The calculations for a single raindrop involves taking into account the laws of physics for the animation, lighting, as well as optical effects, which require intensive computation. When scaled to account for not just one, but the many, many raindrops needed to accurately simulate rain, current computational methods have not been able to provide satisfactory frame rates due to the high computational cost of such a process. In relation to live-action movies: the cost of producing scenes that include simulated weather effects such as rain or snow are also extremely high, requiring multiple takes and high budgets/resources that are often out of the range of small-budget movies[GN06]. By providing a way for users to easily simulate and integrate realistic weather into their productions, hopefully this would cut down production time and smoothen the workflow.

This project will explore the possibilities and attempt to implement a particular proposed method for realistic physically-based rain. Once such an implementation is constructed, we hope to be able to compare and contrast the results with other realistic rain simulation implementations. The ideal conclusion of the study would be to discover and effectively prove a best method for implementing realistic real-time physically-based rain simulations.

1.1 Design Goals

The target Audience for this design project is any individual, firm, corporation, or otherwise entity that may be interested in implementing visually realistic physically-based simulated rain. The results of this design project are to provide an implementation of a realistic rain simulation, based on previous research work by Garg, Rousseau, and NVIDIA. Using this implementation, we hope to compare and contrast different methods of simulating realistic rain, and thus provide a clearer analysis on potential best practices for incorporating realistic rain into various scenes.

1.2 Projects Features and Functionality

The implementation would involve a user interface that would allow the simulation designer to modify certain variables and effectively automate and simplify the production of the rain simulation without requiring a deep technical understanding of the software or process.

2. RELATED WORK

Realistic environmental effects have been a growing topic of study in recent years. In particular, the development of rendering rain realistically has progressed under research projects driven by K. Garg, Pierre Rousseau, and NVIDEA's Sarah Tariq. NVIDEA proposed a novel method of rendering rain using the GPU entirely, which significantly increases the efficiency and speed of the traditional particle system model [Tar07].

In addition to developments in the speed and efficiency of rain rendering, the actual quality of realistic lighting and appearance has also progressed in recent years. Garg and Rousseau in particular have driven research in rendering photorealistic rain in real-time applications that integrate the physical and lighting properties of raindrops [RJG06] [GN06].

Previous research work in rain rendering has been implemented in video games; for example, in Unreal Tournament, Need for Speed, and Microsoft Flight Simulator [RJG06]. However, these methods often involved scrolled textures and geometry centered around the camera, which provide a faster solution at the cost of realism. Past implementations of particle-based systems or computer vision methods have also been used to add rain to still capture or video, although at a high computational cost [RJG06].

3. PROJECT PROPOSAL

The project proposal will include the implementation of simulation software that will use certain proposed realistic physically-based rendering methods for rain, and provide a user interface for a user to be able to modify the parameters of the simulation easily. By supplying this implementation of certain rain algorithms, we can compare and contrast them and hopefully come to a conclusion as to what particular methods may produce better results. This conclusion will hopefully allow our audiences—any individual or organization interested in integrating realistic weather effects into virtual environment productions—to reduce time and resources used to produce such environments.

3.1 Anticipated Approach

The anticipated approach to this project will likely include building a stable platform for the simulation first. Using C++ with Python hooks to integrate with Maya, it would seem practical to build the initial framework such that we could then implement and try different methods of realistically rendering the rain. These methods would likely come from the research papers mentioned in the bibliography.

3.2 Target Platforms

The implementation of this project would take place on the personal computer, likely in a combination of C++ and Python with the use of Maya and Renderman as rendering software. There is also the possibility of directly interacting with the GPU using Direct3D10[Tar07].

3.3 Evaluation Criteria

The evaluation criteria for this project would entail the production and visual comparison of a simulated rendered rain scene using our software to other samples, including that of another implementation by an external group and an actual rain scene shot in the physical world or in a movie.

4. RESEARCH TIMELINE

Project Milestone Report (Alpha Version)

... Completed all background reading (February 1st)

... Implementation of software framework (February 21st)

... Implementation of physically-based rain rendering algorithm proposed in [RJG06] (March 20th)

... Production, comparison and evaluation of rendered scene (March 27th)

... Implementation of rain using algorithm proposed in NVIDEA White Paper [Tar07] (April 20th)

... Production, comparison and evaluation of rendered scene (April 27th)

... Implementation of rain using algorithm proposed in [WWZ*08] (June 20th)

... Production, comparison and evaluation of rendered scene (June 27th)

See Gantt Chart in Figure 2.

Project Final Deliverables

… Computer software system that implements a physically-based rain simulation, with adjustable model parameters for simulation design.

Project Future Tasks

… Incorporate other weather effects aside from rain.

… Allow a mixture of weather effects, perhaps even manipulated by region if possible (snow on the mountains, rain in the valley).

You will fill these sections in as you complete your project for the alpha review and the final document, these sections give psedo-code, charts, images, examples etc to show what you've done over the course of the semester.

We are leaving this section open to creativity too, feel free to add whatever you feel is necessary to relate your

5. Method

6. RESULTS

7. CONCLUSIONS and FUTURE WORK

APPENDIX

A. Optional Appendix

References

[Bar08] MATT BARTON: "How's the weather: Simulating weather in virtual environments", *International Journal of Computer Game Research*, Volume 8, Issue 1, September 2008.

[GN06] K. GARG and S. K. NAYAR. Photorealistic rendering of rain streaks. In ACM SIGGRAPH 2006 Papers (Boston, Massachusetts, July 30 - August 03, 2006). SIGGRAPH '06. ACM, New York, NY, 996-1002.

[RJG06] PIERRE ROUSSEAU, VINCENT JOLIVET, DJAMCHID GHAZANFARPOUR: Realistic real-time rain rendering, *Computers & Graphics*, Volume 30, Issue 4, August 2006, Pages 507-518.

[Tar07] SARAH TARIQ: Rain, NVIDEA White Paper, 2007.

[Tre00] L. A. TREINISH: Multi-resolution visualization techniques for nested weather models. In Proceedings of the Conference on Visualization '00 (Salt Lake City, Utah, United States). IEEE Visualization. IEEE Computer Society Press, Los Alamitos, CA, 513-516.

[WWZ*08] CHANGBO WANG, ZHANGYE WANG, XIN ZHANG, LEI HUANG, ZHILIANG YANG, QUNSHENG PENG: Real-time modeling and rendering of raining scenes, *Visual Computing*, Issue 24, June 2008, Pages 605-616.

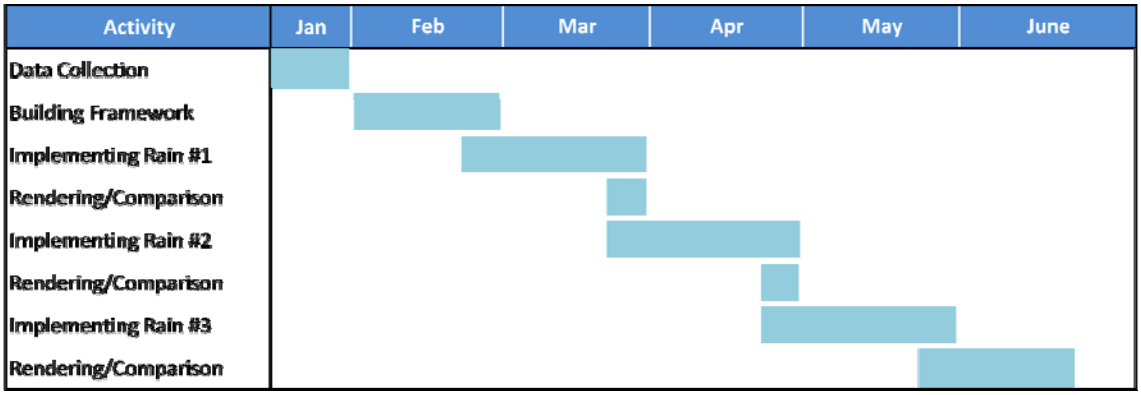

Figure 2: Gantt Chart of proposed timeline.