NISTIR 7431

Visualization, A Tool For Understanding Fire Dynamics

Glenn P. Forney

NIST National Institute of Standards and Technology • Technology Administration • U.S. Department of Commerce

NISTIR 7431

Visualization, A Tool For Understanding Fire Dynamics

Glenn P. Forney
Fire Research Division
Building and Fire Research Laboratory

July 2007

U.S. Department of Commerce
Carlos M. Gutierrez , Secretary

National Institute of Standards and Technology
William Jeffries, Director

Contents

List of Figures

Visualization, A Tool For Understanding Fire Dynamics

Glenn P. Forney

Abstract

Computational tools have been developed at the National Institute of Standards and Technology (NIST) for modeling fire spread and smoke transport in order that various professionals such as fire protection engineers, fire researchers, fire investigators, fire fighters, AHJ's (authorities having jurisdiction) *etc.* may study the dynamics of fire with the ultimate aim of improving fire safety. These models include FDS (Fire Dynamics Simulator) for modeling fire spread and smoke transport and Smokeview for visualizing the resulting fire dynamics. This article will give an overview of many of the features and techniques one can use with Smokeview to better understand fire dynamics. Modeling is especially useful when other methods of study such as experimentation are not practical or are too dangerous to conduct.

1 Introduction

The purpose of fire modeling is to gain a better insight into fire dynamics and how it impacts fire safety not the generation of numbers. Visualization is the primary tool used to accelerate this understanding. This is imperative because of the large amount of data generated by fire models. This article will highlight some of the features that the visualization tool, Smokeview, uses to enable better understanding of fire dynamics.

To illustrate how scientific visualization can be important, consider the Cherry Road case, a 1999 townhouse fire that resulted in line of duty deaths for two fire fighters[1]. NIST was asked by the District of Columbia Fire and Emergency Medical Services Department Reconstruction Committee to examine the fire dynamics of this incident. The Committee had several questions regarding: 1) the injuries that the firefighters had sustained, 2) the lack of thermal damage in the living room where the fallen firefighters were found, and 3) why the firefighters never opened their hose-lines to protect themselves and extinguish the fire. The major source of confusion arose from the fact that the fire fighter farthest from the fire died while the one in the middle (closer to the fire) survived. Figure 1 shows that one dimensional thinking is not always valid. This figure shows temperature contours through the center line of a basement stairwell. The heated gases moved up the basement stairs due to buoyancy and *arched* over the fire fighters. located at the top of the stairs. This visualization makes it clear that the fire dynamics was not one dimensional and that conditions for the middle fire fighter were less hazardous than conditions for the other two.

2 Simulation Overview

NIST, the National Institute of Standards and Technology, has developed a suite of validated computational tools for the simulation and visualization of fire spread and smoke transport. One of the fire modeling tools is called the Fire Dynamics Simulator (FDS)[2, 3]. Developed as a companion to FDS, Smokeview is a scientific visualization tool that converts data to images enabling one to better understand numerically predicted fire dynamics[4, 5]. These tools were developed with an emphasis on ease of use on affordable computer platforms. They may be obtained from http://fire.nist.gov/fds.

FDS predicts smoke and/or hot air flow movement caused by fire, wind, ventilation systems and other factors by solving numerically the fundamental equations governing fluid flow, commonly known as the Navier-Stokes equations. FDS uses a form of computational fluid dynamics (CFD) known as large eddy simulation (LES) to predict the thermal conditions resulting from a fire. LES is a way of describing the effect of turbulence on the flow field. The fire itself is a source term in the governing equations, creating buoyant motion that drives the smoke and hot gases throughout the simulation. The chemistry of the combustion process is complicated by the fact that the fuel for the fire may include room furnishings, ceiling materials, wall, and floor coverings, *etc., i.e.,* a wide assortment of different materials. FDS makes simplifications about the combustion, essentially saying that fuel and oxygen burn readily when mixed. The rate

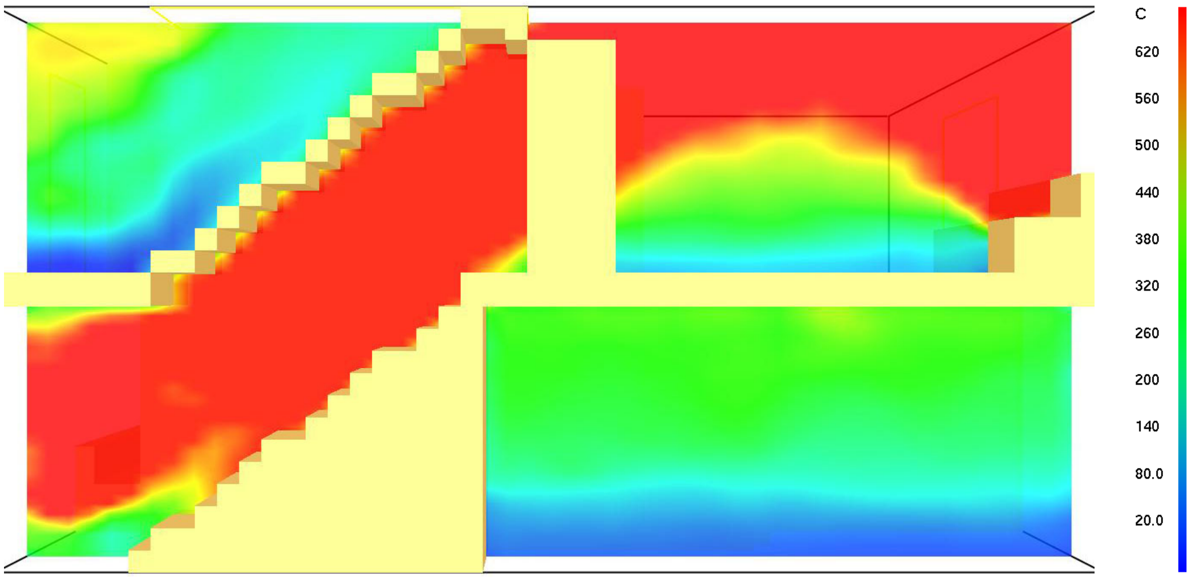


Figure 1: Snapshot of shaded temperature contours through the centerline of a townhouse basement stairway.

at which energy is generated is obtained from experiments. There is no attempt to model the fundamental chemistry which can involve hundreds of chemical reactions.

Both FDS and Smokeview would not have been possible without the recent advent of high-speed computers for performing computations, fast video cards for visualizing results and the Internet for exchanging information and ideas. These programs also would not have been possible without the research needed to develop the underlying fire models and the techniques needed to implement these models accurately and efficiently.

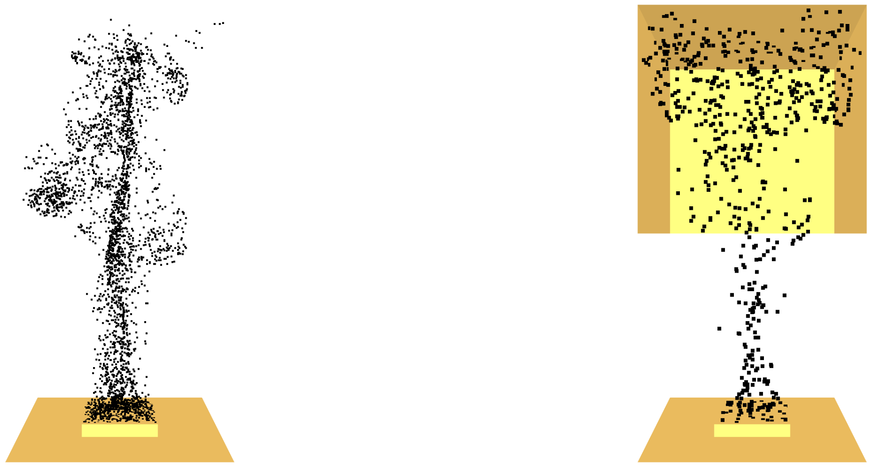
3 Visualization Overview

One of the biggest challenges in visualizing fire dynamics is how to convert the multi-dimensional data generated by a fire model such as FDS into a form that can be easily understood. Fire data can easily have 5 or more dimensions. For example, to display time dependent scalar data would require 5 dimensions: 3 spatial dimensions to visualize position, 1 time dimension and 1 dimension to visualize the variable of interest. Time dependent vector quantities require 8 dimensions to display: 3 spactial dimensions, 1 time dimension, one dimension to visualize the variable as before plus 3 additional dimensions to display the flow direction and speed.

A major challenge to effective visualization is that the computer screen has only 2 dimensions to display these data. A third dimension may be conveyed by rapidly displaying a sequence of images, each image representing a different moment in time. The visualization challenge is even more difficult when conveying results for the printed page, as in this article.

Smokeview visualizes data in two primary ways: quantitative and realistic. Quantitative methods typically map fire modeling data into colors representing a fire modeling variable. Interpreted with a colorbar, one can make quantitative assessments about the data being examined. Some examples used by Smokeview are animated tracer particles, animated two-dimensional slices of gas phase quantities such as temperature or smoke concentration, animated flow vectors, and animated surface conditions such as incident heat flux or burning rates on enclosure surfaces. 3D level or isosurfaces are also used to indicate *where* a particular variable takes on a specified value. Smokeview also visualizes smoke realistically by converting soot density to smoke opacity, with the goal of displaying smoke as it would actually appear to an observer. This technique portrays smoke as it appears visually. Each of these visualization techniques highlight different aspects of the underlying flow phenomena.

Visualization is essential at all stages of the modeling process. It is used before a run to verify the correctness of the scenario geometry, (*e.g.,* locations and size of simulation features), during a run to monitor the simulation (ensuring boundary flows are behaving as intended) and after the run has been completed to analyze the results.



a) Surrounding completely open

b) Upper half of surroundings blocked

Figure 2: Two plume fires visualized using particles. The different fire dynamics for these two cases are not revealed by this visualization method.

Smokeview consists of about 70,000 lines of code. Most of it is written in C using standard libraries such as OpenGL[6] and GLUT[7] for graphics; GD (http://www.boutell.com/gd/), libpng (http://www.libpng.org/pub/png/) and libjpeg (http://www.ijg.org) for generating image files; and libzip (http://www.gzip.org/zlib/) for compression. A portion of Smokeview is written in Fortran 90 to input data generated by FDS. The use of portable libraries allows Smokeview to run on several platforms including computers running Windows, Unix, Linux and OSX. [1]

4 Quantitative Visualization

4.1 Showing Motion

FDS uses particles to simulate water droplets and fuel sprays. One may also introduce particles into a scenario as tracers. All three particle types may be visualized using Smokeview, revealing the underlying flow patterns of the simulation.

Fluid motion may be conveyed by displaying a sequence of still images. A single static particle image, however, is not a good method for showing motion. The two cases shown in Figure 2 both display particles generated by a fire plume. The surroundings in Figure 2a are completely open, while the upper half of the domain in Figure 2b is enclosed. The particle pattern in both cases looks similar though the fire dynamics are quite different.

Streak lines, a new feature of Smokeview version 5, are a good method for showing motion in a static image. A streak line is simply the path a particle takes due to the changing underlying flow field. (If the flow field was unchanging then these *lines* would be called *stream lines*.) The streak lines shown in Figure 3 indicate how particles are affected by the boundary conditions. Streaks are predominantly vertical in Figure 3a since the domain boundary is completely open while the streaks are curved near the top of Figure 3b since the upper half of the domain boundary is blocked. Streak parameters such as streak length and line width may be adjusted using Smokeview's *File/Bounds* dialog box.

A second method for showing motion is the use of animated flow vectors. The vector's color represents the data and the vector's length and direction shows the dynamics of the underlying flow field. Figure 4 shows the fire dynamics of a kitchen fire using both solid shaded contours and a vector plot. Vector plots are better than solid contours for highlighting flow changes, especially in regions where temperatures are uniform.

[1] Any mention of commercial products is for information only; it does not imply recommendation or endorsement by NIST

a) Particles

b) Streaks, 0.5 s long

Figure 3: Two plume fires visualized using streaks. The streak paths show how the presence or absence of an exterior boundary effects the plume flow.

4.2 Assessing Variables

Within the Gas Phase Smokeview allows animated shaded color contours of calculated gas quantities to be drawn at any horizontal or vertical plane in the simulation. To minimize file output, the user specifies the particular slice planes to be visualized. If disk space is not an issue, then the user may specify the entire 3D volume. Smokeview then allows the user to scroll through the 3D volume of data one slice at a time displaying any horizontal or vertical plane. Figure 4b illustrates temperature contours in a vertical plane through the center of a static townhouse kitchen fire (not the Cherry Road case). Regions where the temperatures are below 100 °C are hidden. Hiding unimportant data is a good technique for highlighting data that is important.

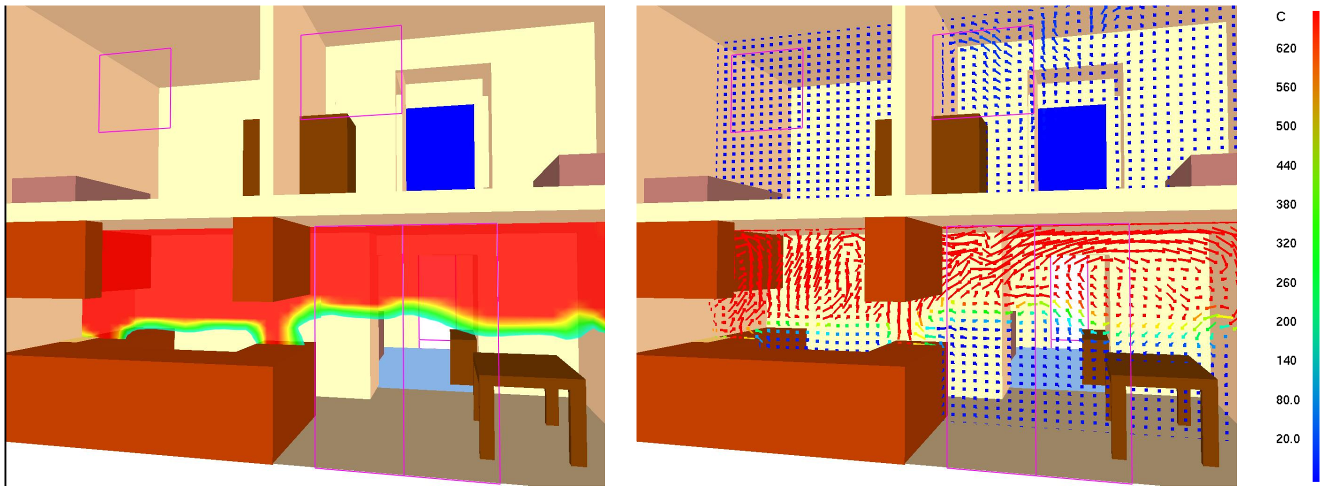
On Surfaces Boundary files contain simulation data recorded at blockage or wall surfaces. Continuously shaded contours are drawn for quantities such as wall surface temperature, radiative flux, *etc*. Figure 5 shows a snapshot of a boundary file animation where the surfaces are colored according to their temperature.

Regions where a surface temperature exceeds its ignition temperature (where burning has occurred) may be colored black. This is also illustrated in Figure 5.

At Particular Locations Smokeview uses isosurfaces to identify *where* a specified level of a gas phase quantity occurs rather than *how much* . For example, FDS uses a mixture fraction model to simulate combustion. In this model, there is a critical or stoichimetric mixture fraction value, such that regions greater than the critical value are fuel rich and regions less than the critical value are fuel lean. Burning then occurs, according to the model, on the level surface where the mixture fraction equals this stoichimetric value. Therefore, it is of interest to visualize these locations.

Another application of isosurfaces is to identify where in the simulation domain a particular temperature occurs. This temperature could represent a hazard or a condition when something happens such as a smoke or heat detector activating. Figure 6 shows the region in a townhouse kitchen fire where the temperature is 100 °C. The time and view point are the same as shown Figure 4.

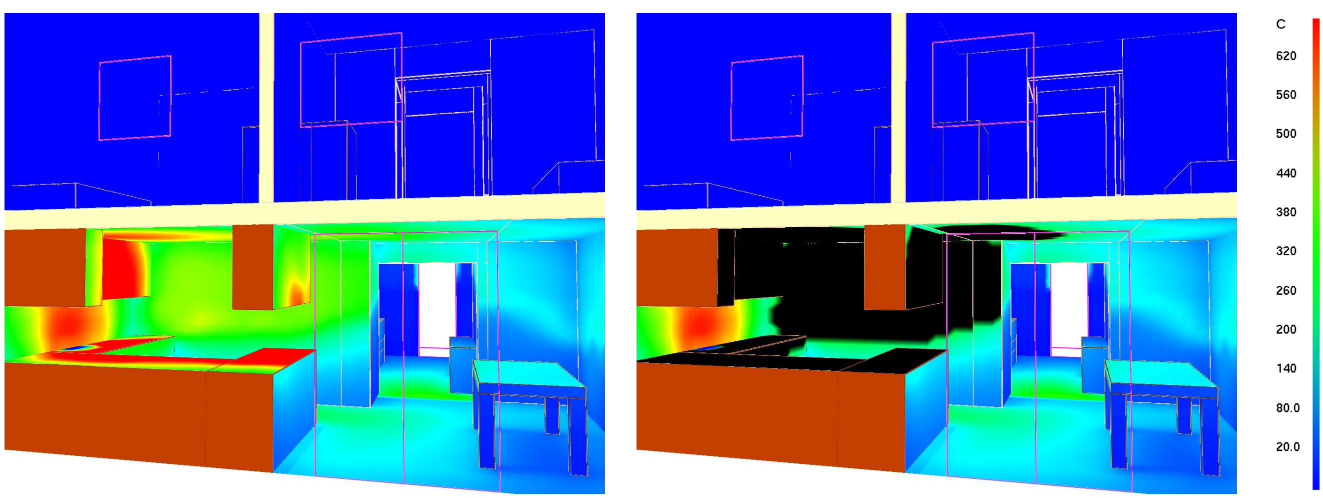
Isosurfaces are generated at each desired time step using a marching cube algorithm[8] modified to remove ambiguities. A decimation procedure is used to reduce the number of resulting triangles by collapsing nodes of triangles with large aspect ratios and re-triangulating. This makes the isosurface look better and also reduces storage requirements.



a) shaded temperature contours

b) shaded 3D temperature vectors

Figure 4: Snapshot of shaded temperature contours and flow vectors through the stove center of a townhouse kitchen fire.



a) temperatures contours

b) temperature contours and ignition regions

Figure 5: Shaded temperature contours on boundary surfaces. The black region in sub-figure b) shows where the surface temperature has exceeded the ignition temperature for that material.

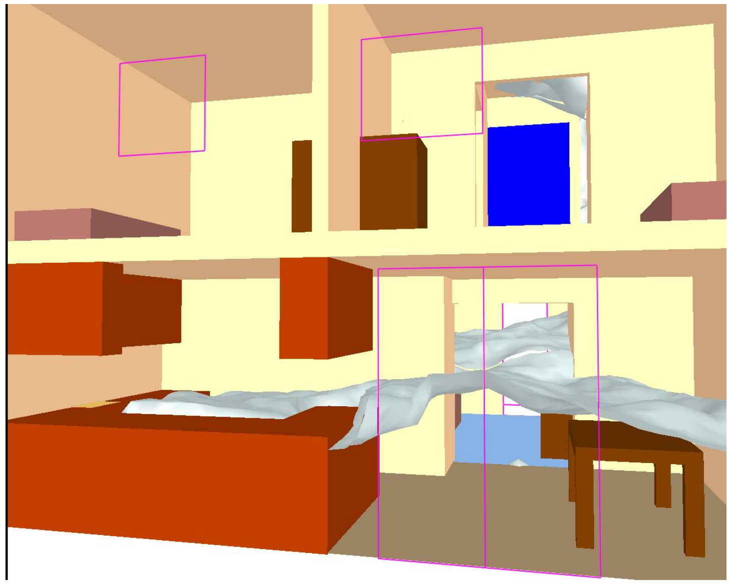


Figure 6: Temperature isosurface at 100 °C.

5 Realistic Visualization

Visualizing smoke realistically is challenging for three reasons. First, the storage requirements for describing smoke throughout the simulation scene at every time step can easily exceed the file size capacities of present 32 bit operating systems which would typically be 2 GB. Second, the computation required both by the CPU and the video card to display each frame can easily exceed 0.1 s, the time corresponding to a 10 frame/s display rate. Finally, the physics required to describe smoke and its interaction with itself and surrounding light sources is complex and computationally intensive. Approximations and simplifications are required.

Smoke visualization techniques described previously, such as the use of tracer particles or shaded 2D contours are useful for analyzing data quantitatively, but are not suitable for applications where realism is required. Some examples of such applications are using Smokeview as a virtual fire fighter trainer or using Smokeview to examine the obscuration effects of smoke. Figure 7 shows smoke and fire displayed realistically.

The approach used by Smokeview for visualizing smoke realistically is similar to that taken in Fedkiw *et. al.*[9] except that interactions with smoke and light are not considered (only the effects of smoke obscuration are visualized). The video hardware is exploited to perform a simple obscuration calculation by using OpenGL to display a series of partially transparent parallel planes. The planes are oriented to be perpendicular to the viewer's line of sight. The transparencies are computed based on physics using data derived from a FDS calculation. Vertices in each plane are colored black or dark grey based on the estimated smoke albedo. The vertices are also assigned an OpenGL α opaqueness parameter. The assigned value depends on the optical smoke thickness, with 0.0 used for completely transparent smoke and 1.0 for completely opaque. Fire is displayed wherever the heat release rate per unit volume exceeds a specified value, otherwise it is assumed that combustion is not occurring and smoke is displayed instead.

6 Summary

Visualization is an important tool for converting the large amounts of data generated by a fire model into a form that one can use to obtain a better understanding of what is being studied. Smokeview uses several techniques for visualizing data, some quantitative and some realistic. Quantitative techniques typically map data to color and screen positions. Realistic techniques in Smokeview use video hardware found on modern computers to convert computed soot densities to opacities which in turn are used to create realistic displays. There is no one best visualization method. The particular method chosen should depend on what aspects of the data is being examined.

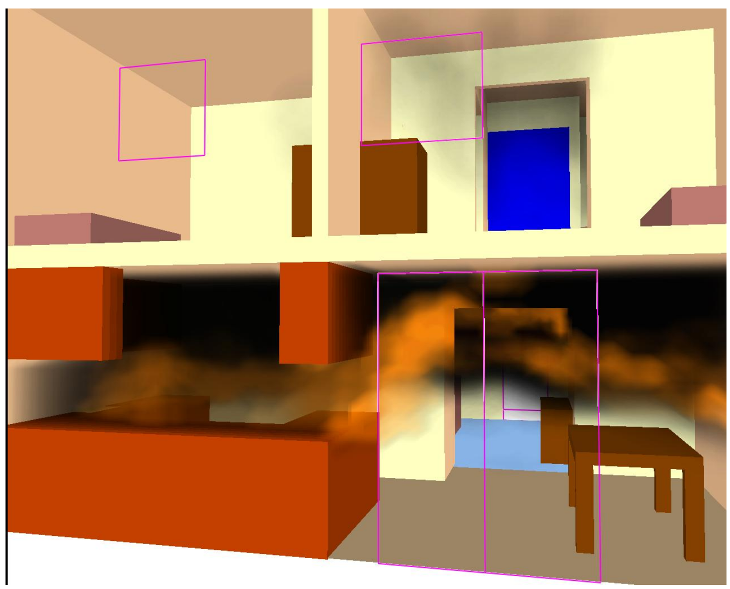


Figure 7: Realistic visualization of smoke and fire using opacities determined from FDS computed soot density.

References

[1] D. Madrzykowski and R.L. Vettori. Simulation of the Dynamics of the Fire at 3146 Cherry Road NE, Washington, DC May 30, 1999. Technical Report NISTIR 6510, Gaithersburg, Maryland, April 2000. URL: http://fire.nist.gov/6510.

[2] K.B. McGrattan, S. Hostikka, and J.E. Floyd. Fire Dynamics Simulator (Version 5), User's Guide. NIST Special Publication 1019-5xxx, National Institute of Standards and Technology, Gaithersburg, Maryland, January 2007.

[3] K.B. McGrattan, S. Hostikka, J.E. Floyd, H.R. Baum, and R.G. Rehm. Fire Dynamics Simulator (Version 5), Technical Reference Guide. NIST Special Publication 1018-5, National Institute of Standards and Technology, Gaithersburg, Maryland, January 2007.

[4] G. P. Forney, D. Madrzykowski, K. B. McGrattan, and L. Sheppard. Understanding fire and smoke flow through modeling and visualization. *Computer Graphics and Applications*, 23(4):6–13, 2003.

[5] G.P. Forney. User's Guide for Smokeview Version 5 - A Tool for Visualizing Fire Dynamics Simulation Data. NIST Special Publication 10175, National Institute of Standards and Technology, Gaithersburg, Maryland, June 2007.

[6] OpenGL Architecture Review Board, Dave Shreiner, Mason Woo, Jackie Neider, and Tom Davis. *OpenGL Programming Guide - The Official Guide to Learning OpenGL, Version 2*. Addison-Wesley, Stoughton, Massachussets, 5 edition, 2005.

[7] Mark J. Kilgard. *OpenGL Programming for the X Window System*. Addison-Wesley Developers Press, Reading, Massachussets, 1996.

[8] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169. ACM Press, 1987.

[9] Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. Visual simulation of smoke. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 15–22. ACM Press / ACM SIGGRAPH, 2001.