Using Layered Manufacturing for Scientific Visualization

Mike Bailey, Oregon State University

Introduction

From the moment each of us was born, we have learned to use our sense of touch to gain knowledge about the world around us. Why should we ever stop doing that? Our sense of touch can be exploited in the understanding of scientific data. Rapid Prototyping, or Layered Manufacturing (LM), is a mainstream technology in engineering product development. It enables engineers to create a prototype part from a new design before entering final production. The Center for Visualization Prototypes (CVP) uses LM technology to create 3D visualization hardcopies, collaborating with scientists to enhance data understanding. During its 10 years of operation, the CVP has made more than 1,000 models for scientists around the world, helping to visualize data in such areas as chemistry, biology, mathematics, geology, astrogeology, cartography, oceanography, engineering, and many others. Along the way, we have learned a lot about using physical hardcopy for understanding science and about turning various types of data into physical form.

Typical large-scale manufacturing is characterized by *subtractive* processes, for example, starting with a block of material and then removing pieces of the material with a milling machine until only the desired part remains. Layered Manufacturing, on the other hand, is characterized by *additive* manufacturing processes. They start with nothing, and add material layer-by-layer or drop-by-drop until the 3D part is completed.

It is not obvious at first, but there are significant advantages to additive manufacturing for scientific visualization:

- Extremely complex parts can be handled. A subtractive manufacturing element (e.g., the milling machine bit) doesn't have to try to snake its way through complex geometry, likely gouging the part along the way.

- Parts can be started and let run to completion without further intervention. They do not need to be flipped over or otherwise reoriented in the middle of the process.

- Topologically impossible 3D parts can be made. Geometry that is topologically impossible in 3D is not usually impossible 2D-at-a-time. Already-assembled interlocking parts, such as chains and ball-in-socket joints, can be produced.

There are many Layered Manufacturing technologies and vendors [1,2] The CVP is using a Z Corporation Z406 machine [3] that selectively applies a bonding agent to a very thin layer of powder. The Z406 can also apply color to the model's outer surface. This is very important to visualization applications in that parts not only convey shape, but can also map scalar data values to the surface.

The Model Input File: Creation, Preview, and Repair

The *de facto* standard for all LM machines is the *STL* file format. This has been useful for users like us who want to make considerable investments in software tool development, and also want to be reassured that those tools will work on many different devices and will continue to work well into the future.

An STL file consists of a very simple format – it describes 3D solids by listing the triangles that bound their exterior, shown below in Table 1:

```
solid
  facet normal    0.121    0.380   -0.917
    outer loop
      vertex    1.5000    1.8882    0.0511
      vertex    1.5000    2.2500    0.2010
      vertex    1.9302    1.9783    0.1450
    endloop
  endfacet
  facet normal    0.175    0.175   -0.969
    outer loop
      vertex    1.9302    1.9783    0.1450
      vertex    1.8882    1.5000    0.0511
      vertex    1.5000    1.8882    0.0511
    endloop
  endfacet

       . . .

endsolid
```

Table 1: STL File Sample

The STL format is often described as a “bucket of triangles”, because, while this is an easy format to create and an easy format to write, it lacks much of the robustness that geometric modeling practitioners have come to expect. For example, there is no guarantee that the set of triangles form a legal solid, or that there are no T-intersections, or that the triangle vertices are all oriented consistently, or that the surface normal vectors are all outward-facing. We have found that many STL files that even originate from robust solid modeling CAD systems are not always correct. STL files that are generated from scientific research software are even less likely to be correct. Thus, before fabrication technology can be used productively, better ways had to be found to deal with the STL file format.

The CVP project developed a program to preview and repair STL files [2]. We found it useful to be able to:

- Import ASCII and binary STL files
- Import color STL files using a CVP ASCII color STL extension
- Interactively transform in 3D to inspect the part
- Display the part as points, lines, reduced lines, shaded surfaces, or shrunk triangles
- Scale and re-orient parts
- Detect and fix model cracks and reversed triangles
- Use color to show difficulty of mold creation in different orientations
- Display parts using 3D ChromaDepth
- Export parts in STL, color STL, or PLY (used by the color Z Corp machine)

The most important aspect of this effort was to turn the “bucket of triangles” into a robust winged-edge adjacency data structure. It does this by accumulating unique vertices in a balanced tree upon reading the STL file. Once the vertices in common have been determined, common triangle edges are located, which then also gives the triangle adjacency information.

In scientific data, STL flaws are usually of two types: (1) surface normals pointing in the wrong directions, and (2) missing triangles that form cracks in the surface. There are two stages to fixing the normals. The first stage divides the triangles into a Binary Space Partition (BSP) structure. The second stage processes each triangle by drawing an infinite ray from the center of the triangle. It intersects that ray with each of the BSP boxes. If it hits a box, then every triangle in that box is added to a list to be checked for collisions. In order to make sure it doesn't check the same triangle twice, a flag is set in each triangle after it has been tested. If the ray hits an odd number of faces behind it, and an even number of faces in front of it, then it was originally facing outward, which is correct. If not, the normal was facing the wrong direction and it is flipped.

Cracks are found by looking for edges that only have one triangle attached to them. These edges are placed into a list. It then looks for two edges in this list with a point in common. When it finds these two edges, it completes the third edge between them to create a new crack-patching triangle.. If that newly-created edge does not complete another triangle, then it is added to the bounds-a-single-triangle list. This list continues to be processed until it is empty. These methods alleviate almost all of the problems that typically arrive with other people's STL files. Because of this, we can accept almost any surface data that a research project wants to throw at us.

Colored Parts

Color is a mainstay in visualization data display. The good news is that some LM devices, including the Z406 used here, are capable of making parts in color on a triangle-by-triangle basis. The bad news is that the STL file format does not allow for *any* color specification. To get around this, the Z406 expects color parts to be specified in a different polygon format called PLY. This, however, does nothing to help applications such as ours that are tailored to produce STL files. So, to make color specifiable from existing STL-based applications, we added our own extension to STL. In our version of STL, a line of the form:

```
color r g b
```

can be included anywhere in the STL file and can be included as many times as needed. The r, g, and b values are in the range (0.,1.). This line has the effect of coloring all triangles after this line to (r,g,b), until the next `color` command is encountered. This allows us to preview color parts and then export them as a PLY file. This ability is useful in scientific visualization because it allows us to display additional scalar information on the model surface. For example, Figure 1a-d below shows a Mars globe with surface coloring, a molecule that shows electric charge, a terrain visualization of the state of Oregon with a satellite photo attached, and 3D mathematical surfaces showing various types of curvature.

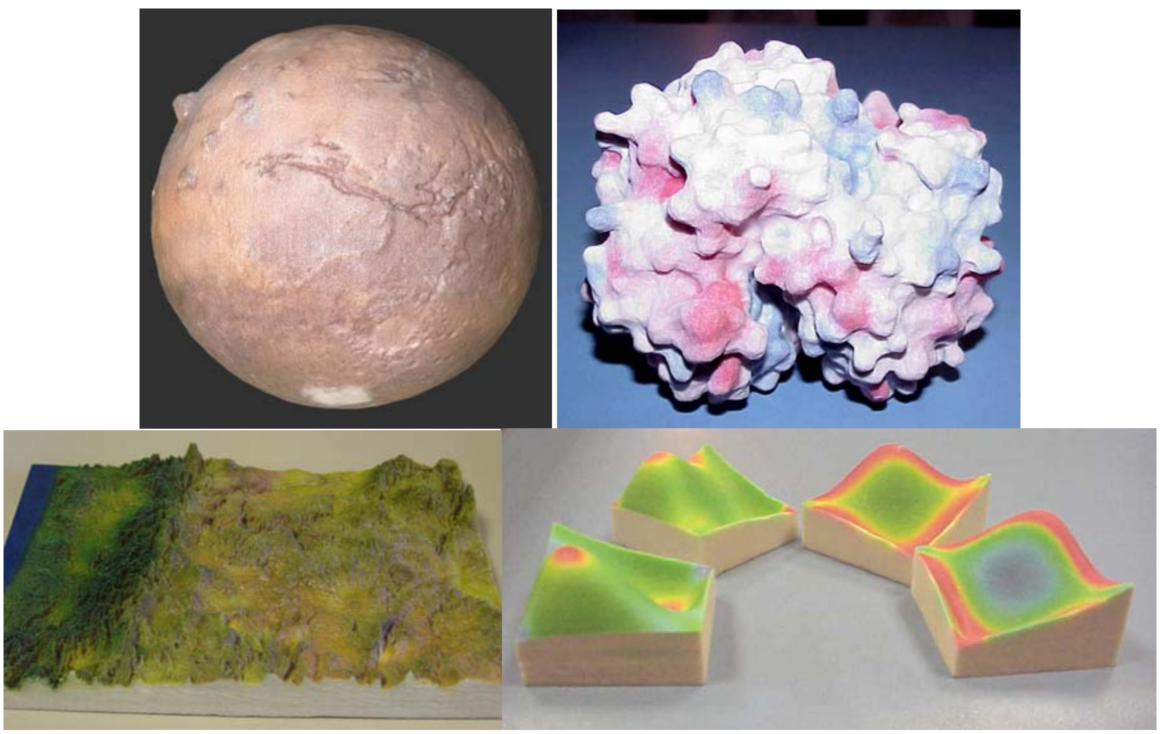

Figure 1: Examples of using color to enhance models
1a: Colored mars globe, 1b: Hemoglobin molecule showing electrical charges, 1c: Terrain Map of Oregon with Superimposed satellite image, and 1d: different mathematical curvatures on identical surface shapes

Three Categories of Models

In our experience, visualization physical models fall into three categories, according to how they will be used: (1) for viewing, (2) for touching, and (3) for collision.

Useful Because of Viewing

One of the best attributes of having 3D physical models in visualization is that they are portable and can be taken anywhere to be shown to anyone. It is relatively straightforward to perform 3D graphics on a display monitor, and even somewhat straightforward to apply stereographics to it, but that only works when you are there at the monitor. A physical model can act as a portable stereographics display. It can also be put on display for all of us to enjoy and learn from. Along these lines, Figure 2a shows a visualization of the October 2003 Southern California fires. Figure 2b shows a USA map that is now in the US Library of Congress map collection:

Useful Because of Touch

Many 3D shapes are complex enough that they defy complete understanding, even on a 3D stereographics display system. A physical model allows users to touch, hold, rub, poke, pinch, rotate, and zoom the shape as a way of better understanding it. Figure 2c shows part of an anthrax molecule, produced for a drug-search project. There are a number of interesting nooks and crevices, all of which are much more understandable after running one's fingers over them.

Useful Because of Collision

We have found that physical models are especially good when characterizing physical collision needs to be part of the understanding. A good example of this is molecular interaction. While good procedural collision detection algorithms exist, many non-convex points of contact give them a hard time. Also, even when detected, haptic systems have a difficult time completely replicating the force produced from multiple points of contact. Physical models finesse the issue. Figures 2d and 2e show parts of the Black Beetle Virus being fit together. [5,6] These particular models enabled biomolecular researchers to gain new insights into the docking and structural relationships of these components of the virus.

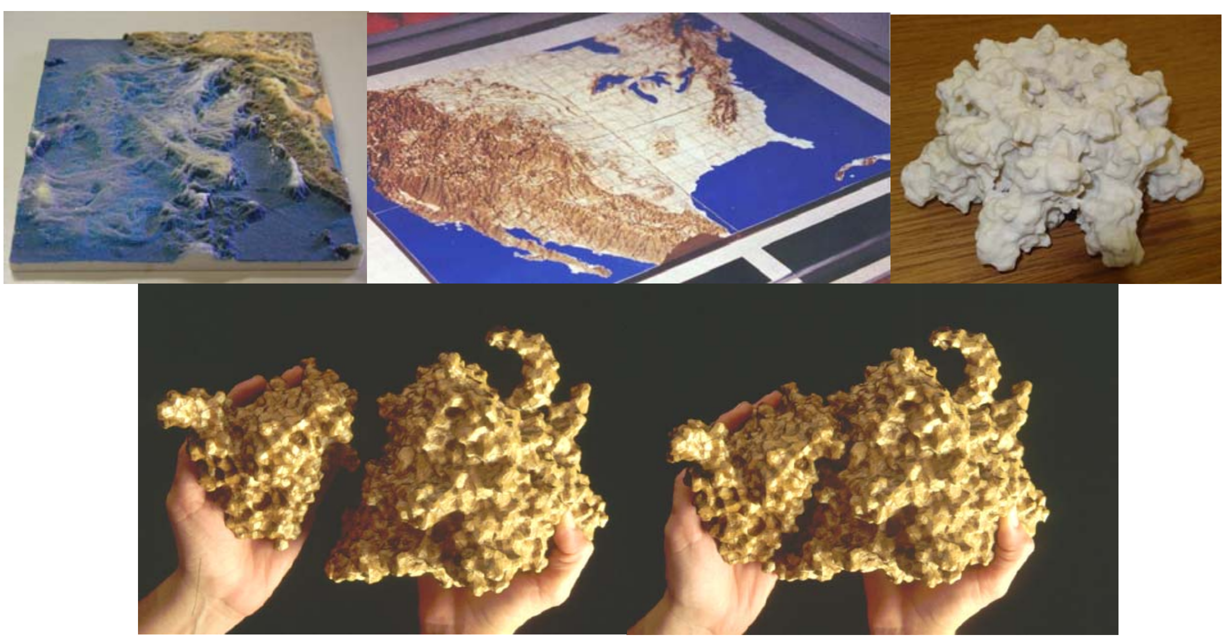

Figure 2: Different Uses for Visualization Hardcopy. From top-to-bottom, left-to-right: Brushfire visualization, USA map, Anthrax, molecular docking

Converting Volume Data to Physical Models

When requesting a graphical isosurface, a single scalar value, S^* is given. A manufacturable isovolume must be a legal solid, which means that it must be continuously bound on all sides. In requesting an isovolume, two scalar values, S_{min} and S_{max} must be specified.[1] Turning these two isovalues into a legal solid is a two-step process:

1. Compute each isovalue's corresponding isosurface
2. At the boundaries of the volume, cap the gap between the isosurfaces.

Much of the current graphics and visualization literature is concerned with polygon *decimation*. Polygon decimation seeks to eliminate detail that is perceptually unnecessary, in order to achieve better graphics performance. This works well for computer graphics where displays can exploit smooth shading and dynamics tricks to get away with less detail. But, physical solids can play no such trickery. Large polygons that look smooth on a graphics display will create a fabricated surface that looks coarse and "chunky". Fortunately, when fabricating isovolumes, display speed is not relevant. Whereas interactive graphics encourages the trading of display quality and accuracy for speed, fabrication encourages maximum quality display regardless of polygon count. We use the term *triangle incremation* to describe the adding of such scene detail by adaptive subdivision [7]. Figures 3a, 3b, and 3c show three models that resulted from volume data. [8]

Slope Enhancement

In everyday dealing with the world around us, we gather considerable information about an object's shape by noticing its interaction with light. In the case of Lambertian shading, the more a surface's perpendicular normal vector points towards the light source, the brighter the surface will be. Figure 3d shows a model of the Grand Canyon. There is obviously a different light intensity on the walls of the canyon than on the nearly horizontal surface above it. But, for visualization, this is not enough. We found that we needed to exaggerate that intensity difference to enhance the understanding of the model. In Figure 3e, the intensity has been raised to a power, resulting in a faster intensity drop-off with slope. The steep walls of the canyon are more obvious. This also enhances subtler features such as river tributaries and mesas. Figure 3f shows the super-shaded terrain with a satellite image attached.

[1] If just the inside or just the outside is desired, the value of S_{min} can be set to $-\infty$ or the value of S_{max} can be set to $+\infty$.

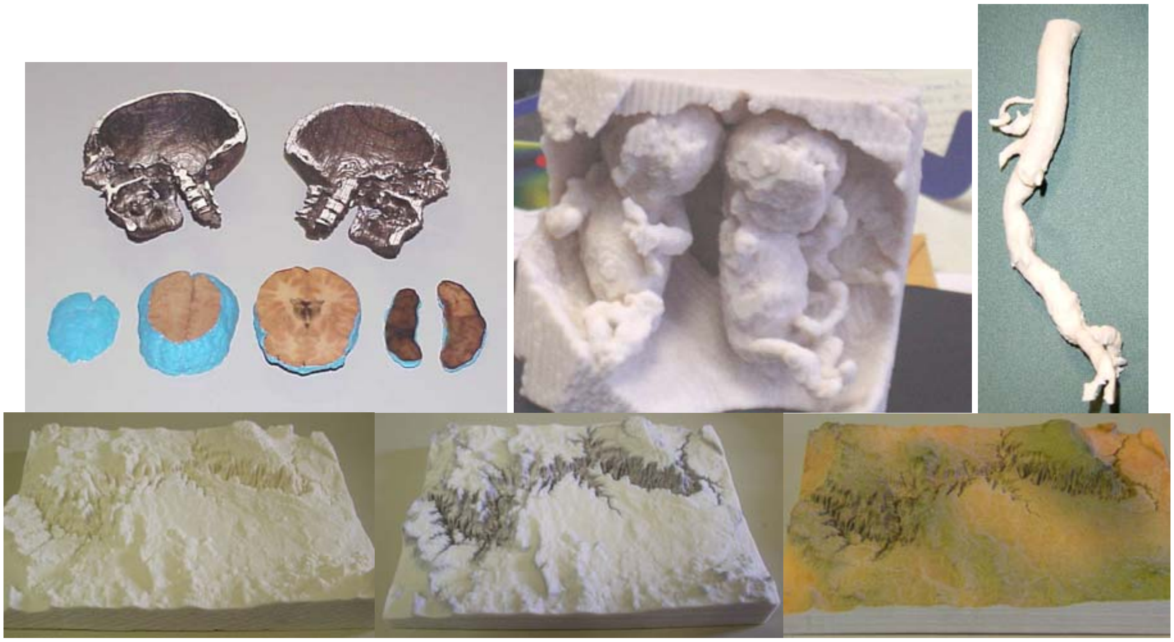

Figure 3: Visualization Hardcopy Enhancement Techniques. (3a) human head, (3b) human fetuses from 3D ultrasound, (3c) aortic aneurysm from CAT scan slices, (3d) unmodified Grand Canyon terrain model, (3e) slopes enhanced, (3f) slopes enhanced and satellite image added

Education and Outreach

We have used layered manufacturing to create physical models to aid education at both the K-12 and undergraduate levels. One of the most interesting experiments was using our physical models in a 9th grade Earth Science class. The teacher used a model of the United States topography to show her students how the shapes of mountains give a clue to their age. They compared the Rockies and the Appalachians, both in terms of appearance and feel. Now it's your turn: which is older, the Rockies or the Appalachians? Determine your answer, and then see the footnote below for the answer.[2]

Conclusions

As children, we relied heavily on seeing and touching the 3D world around us to explore and understand. As adults, nothing has changed except that we wish to explore abstract scientific data as well. The production of 3D visualization hardcopies has been very successful as research aids and as teaching tools. But, we have discovered that the production of physical models from scientific data is not as straightforward as one would like, and thus has required the development of a number of tools and "tricks" to convert general 3D data into physical model form. We have been pleasantly surprised how well this has been received among the scientists and engineers with whom we have collaborated, and how insightful these models have been.

Acknowledgements

This project gratefully acknowledges funding from the National Science Foundation under grant 9809224. The CVP also acknowledges former graduate students Dru Clark and John Rapp for their work on this project. Many thanks to the collaborators whose data is shown here: Mike Malin (Mars); Graham Richards and Meir Glick

[2] The Appalachians have less elevation and are more rounded on the tops than the Rockies, indicating they have had more time to erode, and thus were formed longer ago. Seeing this is interesting, but feeling it makes a more lasting educational impression.

(anthrax), Jack Johnson (black beetle virus), Tom Nelson (3D ultrasound); and Anne-Virginie Salsac, Juan Lasheras, and Steve Sparks (aneurysm). Thanks to Anna Wilder for the use of her 9th grade Earth Science class. Many thanks to Mike Gannis for his article review comments.

References

[1] M. Burns, *Automated Fabrication*, Prentice-Hall, 1993.

[2] S. McMains, "Introduction to Layered Manufacturing Technologies", *Communications of the ACM*, THIS ISSUE.

[3] http://www.zcorp.com

[4] M. Bailey, J. Rapp, and D. Clark, "A Portable Graphics Program for Rapid Prototyping File Display, Repair, and Interchange ", *Journal of Tsinghua University, Vol. 42, No. S2, Proceedings of the Second Annual International Conference on Rapid Prototyping,* Beijing, China, August 19-20, 2002, pp. 167-174.

[5] M. Bailey, K. Schulten, and J. Johnson, "The Use of Solid Physical Models for the Study of Macromolecular Assembly," *Current Opinion in Structural Biology,* Vol 8, No 2, April 1998, pp 202-208.

[6] K. Svitil, "A Touch of Science," *Discover Magazine,* Volume 19, Number 6, June 1998, pp. 80-84.

[7] M. Bailey, "Manufacturing Isovolumes", *Proceedings of the International Workshop on Volume Graphics*, Swansea, UK, March 24-25, 1999, pp. 133-146.

[8] D. Nadeau and M. Bailey, "Visualizing Volume Data Using Physical Models", *Proceedings of IEEE Visualization 2000*, October 2000, pp. 497-500.

EXTRA FIGURES THAT COULD BE USED ELSEWHERE IF SPACE PERMITS: