FRONTISPIECE.   Lime Jell-O

RAY TRACING JELL-O BRAND GELATIN

New technology is presented for imaging a restricted class of dessert foods.

PAUL S. HECKBERT

Ray tracing has established itself in recent years as the most general image-synthesis algorithm [10]. Researchers have investigated ray-surface intersection calculations for a number of surface primitives. These have included checkerboards [Whitted 80]; chrome balls [Whitted 80]; glass balls [Whitted 80]; robot arms [Barr 82]; blue abstract things [Hanrahan 82]; more glass balls [Watterberg 83]; mandrills [Watterberg 83]; more mandrills [Sweeney 83]; green fractal hills [Kajiya 83]; more glass balls [SEDIC 83]; aquatic blobby things [Kaw 83]; more chrome balls [Heckbert 83]; pool balls [Porter 84]; more glass balls [Kajiya 86].

Unfortunately, *nobody* has ray traced any food. So far, the most realistic foods were Blinn's classic orange and strawberry images, but these were created with a scan-line algorithm [2]. The *Dessert Realism Project* at Pixar is addressing this problem. This article presents new technology for ray tracing a restricted class of dessert foods, in particular Jell-O®-brand gelatin. We believe this

Jell-O is a registered trademark of General Foods.

method may have application to other brands of gelatin and, perhaps, pudding as well.

This article is divided into three parts: methods for modeling static Jell-O, simulation of Jell-O motion using impressive mathematics, and ray-Jell-O intersection calculations.

JELL-O SHAPE
To model static Jell-O, we employ a new synthesis technique wherein attributes are added one at a time using abstract object-oriented classes we call *ingredients*. Ingredient attributes are combined during a pre-processing pass to accumulate the desired set of material properties (consistency, taste, torsional strength, flame resistance, refractive index, etc.). We use the RLS orthogonal basis (raspberry, lime, and strawberry, as shown in the figure below, from which any type of Jell-O can be synthesized [9].

Ingredients are propagated through a large 3-D lattice using vectorized pipeline SIMD parallel processing in a systolic array architecture that we call the *Jell-O Engine*. Furthermore, we can compute several lattice points simultaneously. Boundary conditions are imposed along free-form surfaces to control the Jell-O shape, and the ingredients are mixed using *relaxation* and *annealing* lattice algorithms until the matrix is chilled and *ready-to-eat*.
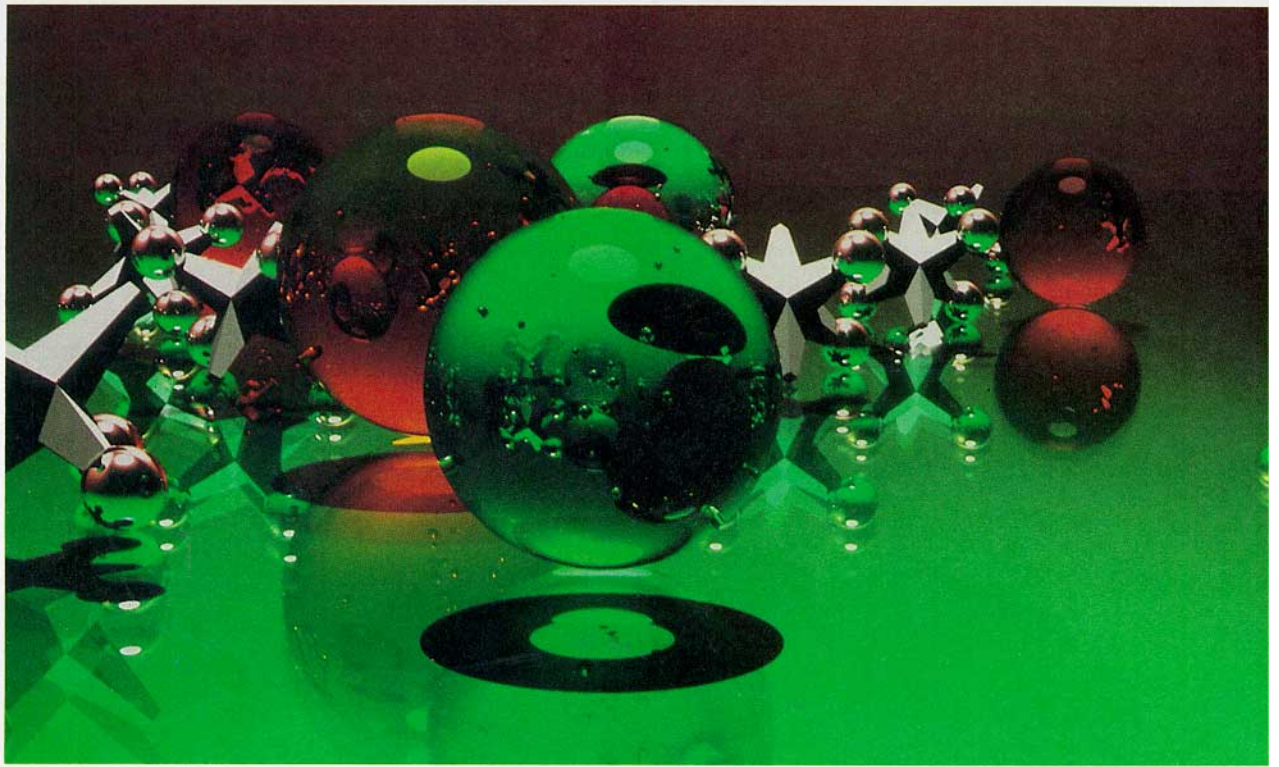
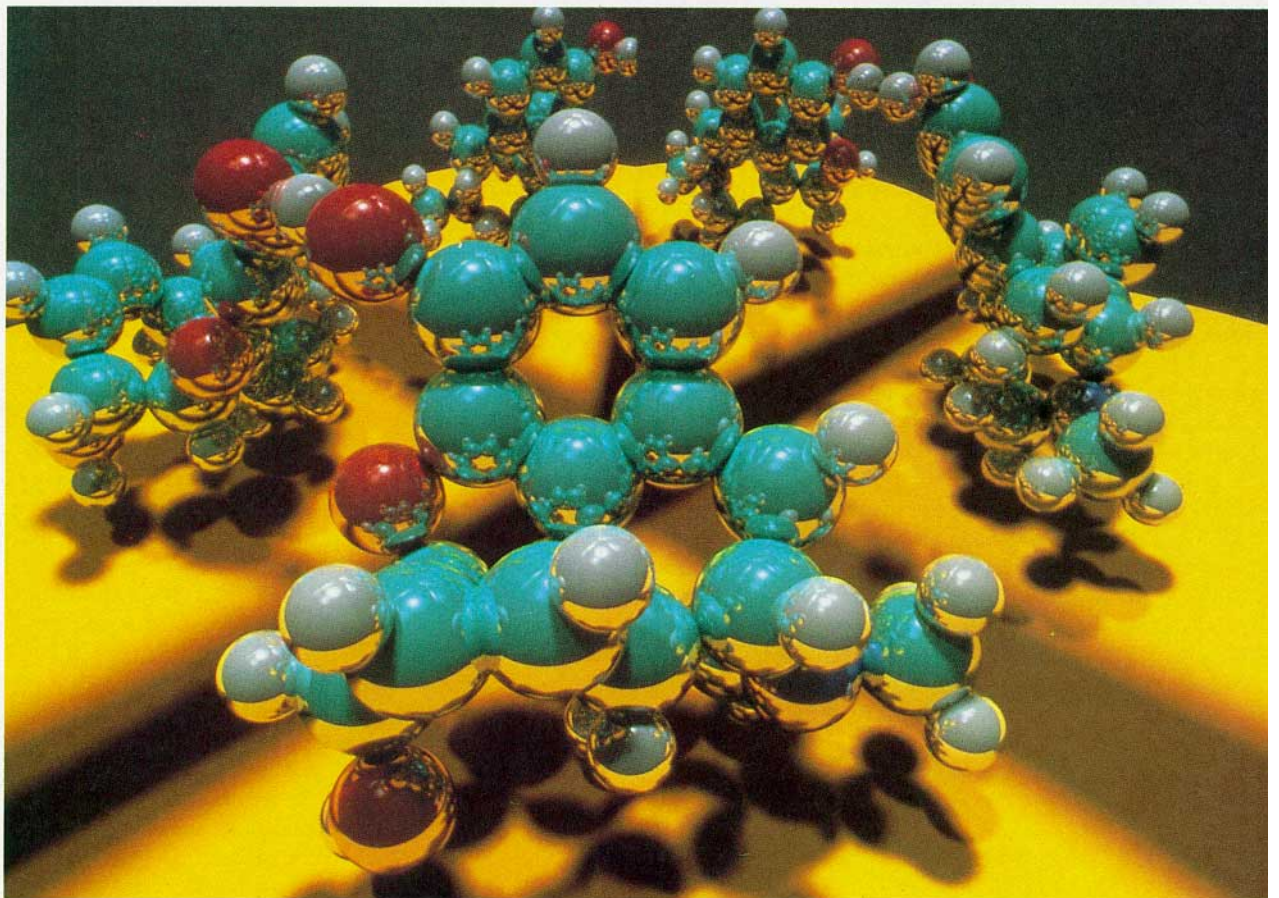FIGURE 1. Glass Balls (G. Ward, 1986)

FIGURE 2. Chrome Balls (P. Heckbert, 1983)

JELL-O DYNAMICS

Previous researchers have observed that, under certain conditions, Jell-O *wiggles* [8]. We have been able to simulate these unique and complex Jell-O dynamics using spatial deformations [1] and other hairy mathematics. From previous research with rendering systems, we have learned that a good dose of gratuitous partial differential equations is needed to meet the paper quota for impressive formulas.

Therefore, we solve the Schrödinger wave equation for the Jell-O field **J**:

$$\nabla^2 \mathbf{J} + \frac{2m}{\hbar}(E - V)\mathbf{J} = 0.$$

Transforming to a spherical coordinate system [7],

$$\nabla \mathbf{J} = \xi_x \frac{\partial \mathbf{J}}{\partial r} + \xi_y \frac{1}{r}\frac{\partial \mathbf{J}}{\partial \theta} + \xi_z \frac{1}{r \sin \theta}\frac{\partial \mathbf{J}}{\partial \phi}$$

$$\nabla^2 \mathbf{J} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial \mathbf{J}}{\partial r}\right) + \frac{1}{r^2 \sin \theta}\frac{\partial}{\partial \theta}\left(\sin \theta \frac{\partial \mathbf{J}}{\partial \theta}\right) + \frac{1}{r^2 \sin^2 \theta}\frac{\partial^2 \mathbf{J}}{\partial \phi^2}.$$

Fuller has given a concise and lucid explanation of the derivation from here:

> The "begetted" eightness as the system-limit number of the nuclear uniqueness of self-regenerative symmetrical growth may well account for the fundamental octave of unique interpermutative integer effects identified as plus one, plus two, plus three, plus four, as the interpermuted effects of the integers one, two, three, and four, respectively; and as minus four, minus three, minus two, minus one, characterizing the integers five, six, seven, and eight, respectively [3].

In other words, to a first approximation:

	J = 0.	
	The Jell-O* Equation	

RAY-JELL-O INTERSECTION CALCULATION

The ray-Jell-O intersection calculations fortunately require the solution of integral equations and the simulation of Markov chains [6], so they cannot be computed efficiently. In fact, we have proved that their solution is linear-time reducible to the traveling-salesman problem, where n is the number of Jell-O molecules, so we can be sure that ray tracing Jell-O will be practical only on a supercomputer [5].

FIGURE 3. Members of the Dessert Foods Division, Pixar: from left, H. B. Siegel, Paul Heckbert, Charlie Gunn, and John Bradstreet

FIGURE 4. Gunn analyzing a Jell-O Sample

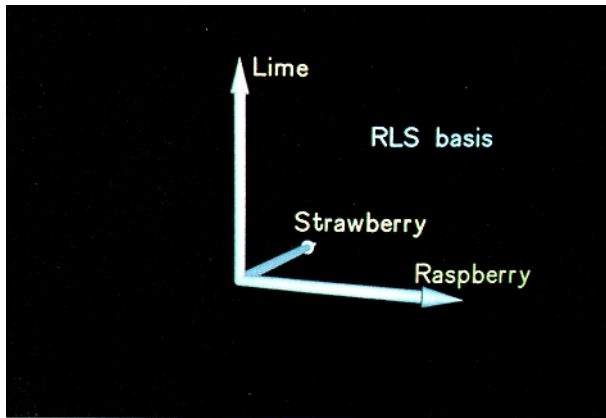FIGURE 5. Jell-O Ingredients (© 1986 Palm Press)
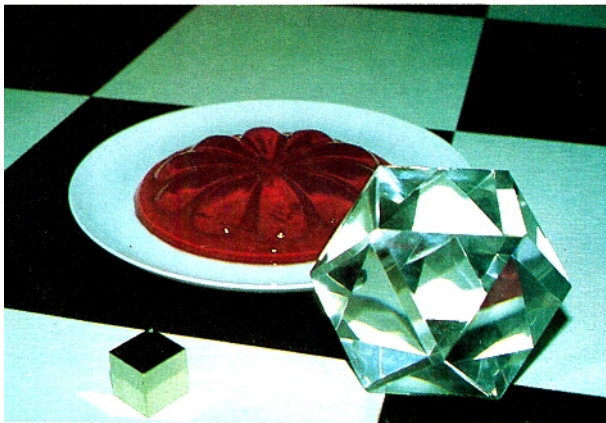
FIGURE 6. The Jell-O Coordinate System

FIGURE 7. Preliminary Results from the Jell-O Engine

IMPLEMENTATION

A preliminary implementation has been completed on a VAX 11/780 running the UNIX® operating system. To create a picture using the full Jell-O Engine simulation, we estimate that 1 CPU eon of CRAY time and a lot of hard work would be required. We made several simplifying approximations, however, since the article is due today. As a first approximation, we have modeled a gelatin cube governed by the first-order Jell-O equation with judiciously selected surface properties; that is, color = (0, 255, 0). The frontispiece for this article was created with this model.

Work is underway on a complete Jell-O Engine implementation in Lisp *flavors*. We will shortly begin computing a 100-by-100 image of a bowl of lime Jell-O using a roomful of Amigas [4]. The picture should be ready in time for SIGGRAPH with hours to spare.

UNIX is a registered trademark of AT&T Bell Laboratories.

CONCLUSIONS

Jell-O goes well with a number of other familiar objects, including mandrills, glass balls, and teapots. The composition and animation possibilities are limited only by your imagination (personal communication by Lance Williams, 1980). The Dessert Foods Division is generalizing the methods described here to other brands of gelatin. Future research areas include the development of algorithms for ray tracing puddings and other dessert foods. Another outstanding problem is the suspension of fruit in Jell-O, in particular, fresh pineapple and kiwifruit.

Jell-O is:
- visually appealing
- futuristic
- hydrodynamically captivating
- tasty
- goes well with other objects.

REFERENCES
1. Barr, A.H. Ray tracing deformed surfaces. *SIGGRAPH 86 Proc. 20*, 4 (Aug. 1986), 287–296.
2. Blinn, J.F. Computer display of curved surfaces. Ph.D. thesis. Computer Science Dept., Univ. of Utah, Salt Lake City, 1978.
3. Fuller, R.B. *Synergetics.* MacMillan, New York, 1975, p. 125.
4. Graham, E. Graphic scene simulations. *Amiga World* (May–June 1987), 18–95.
5. Haeberli, P., and Heckbert, P. A Jell-O calculus. *ACM Trans. Graph.* (special issue on ray tracing moist surfaces). Submitted 1872. To be published.
6. Kajiya, J.T. The rendering equation. *SIGGRAPH 86 Proc. 20*, 4 (Aug. 1986), 143–150.
7. Plastock, R.A., and Kalley, G. *Schaum's Outline of Computer Graphics.* McGraw-Hill, New York, 1986.
8. Sales, S. *The Soupy Sales Show*, 1966.
9. Weller, T. *Science Made Stupid.* Houghton Mifflin, Boston, Mass., 1985.
10. Whitted, T. An improved illumination model for shaded display. *Commun. ACM 23*, 6 (June 1980), 343–349.

Author's Present Address: Paul S. Heckbert, Dessert Foods Division, Pixar, San Rafael, CA 94913-3719.