

Isoluminance Contours for Animated Visualization

Marion Cottingham

The University of Western Australia, Australia

EVOLUTION OF THE ISOLUMINANCE CONTOURS MODEL

Visualization software applications were introduced to enable the sheer volume of data produced by computers worldwide to be viewed in picture form. Visualization makes it easier for the human visual system to pick up trends, patterns, and relationships in the data. The first wave of these applications typically generated simple two-dimensional line graphs, bar charts, and pie charts on expensive graphics workstations.

The second generation enabled three-dimensional images, with single data points typically denoted by dots or crosses, as shown in Figure 1. Axes lines or boxes enclosing the data were typically displayed. The whole image could then be rotated with the data points to give some indication of their orientation to each other. Since the majority of computers were not fast enough to perform the rotation in real-time, there was often some confusion in the picture's interpretation.

Graphics software applications became available, such as OpenGL or DirectX, containing functions for creating simple solid objects such as cubes, cylinders, cones, and spheres, and researchers could now attempt their own visualization. Most of these graphical applications were based on the polyhedral representation and did not pro-

vide any control over the number of patches representing each solid. Being able to animate the data points in real-time enhances trends and relationships in the data that are not easily identified in a still image. Since time is crucial in animation, representing solids with the minimum number of patches is of utmost importance.

The Isoluminance Contour model (IC model) was introduced in 1980 (Cottingham, 1981) as a means of representing these simple solids. It provides realistic results that far outweigh those produced using traditional methods, and in a fraction of the time. The amount of data used to generate an image can be easily adapted to suit each frame making real-time animation possible even on slow PCs.

Generating Images Using Traditional Graphics Techniques

Traditional graphics techniques adopt a divide-and-conquer approach to rendering solid objects. Each solid's surface is subdivided into planar patches that are generated one-by-one to produce the image while balancing speed with realism.

Storage Considerations

Each surface patch consists of four edges, each defined by two vertices specified in three-dimensional space. Topological information defines how edges and vertices are connected. Because surface patches from the simple solids form complete rows and columns, the topology can be implied by the order of storage using the Compressed Data Structure (CDS) (Cottingham, 1985; 1987).

Figure 2 shows a sphere represented by 640 facets, stored in 41 rows of 17 columns and generated by drawing lines to connect vertices stored at adjacent columns and rows. This is termed a wire frame drawing.

For speed, wire-frame drawings with only the edges of patches drawn are often used for quick previewing, but these do not provide much realism. At the next level, patches are flat shaded by calling a polygon fill function. However, in general, only triangular surfaces patches are guaranteed to be planar, so it is common practice for surface patches to be triangulated in an attempt to im-

Figure 1. Data cloud with data points denoted by crosses

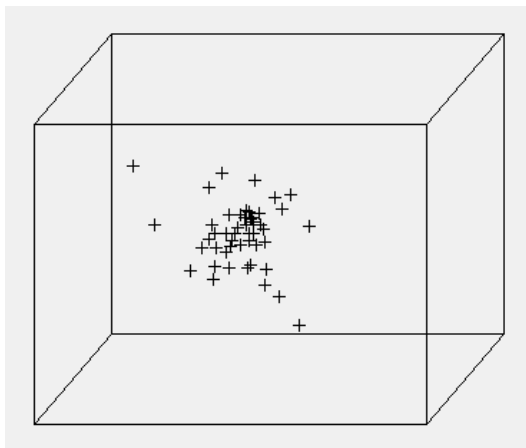
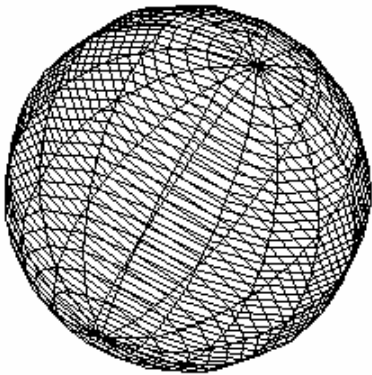


Figure 2. Wire-frame sphere



prove the degree of realism—this doubles the number of patches. Figure 3 shows a sphere defined by 11 rows and 21 columns of vertices that specify 400 triangular patches or 400 quadrilateral patches.

Isoluminance Contour Model (IC Model)

The Isoluminance Contour (IC) model takes an entirely different approach by representing solids by planar contours that are defined by a collection of perimeter points. Points that share the same orientation to the light source are chosen, therefore, they all share the same constant chromaticity. To generate a contour, the illumination model needs to be called only once before calling a polygon fill function. Contours are rendered in depth order. For speed, you can also simply apply linearly spaced chromaticity values without calling an illumination model at all and select one that achieves the required effect.

Using the Isoluminance Contour model, each contour is defined by a single row of vertices from the CDS array, so there is no need to connect vertices from adjoining rows. Figure 4 shows the planar contours drawn as a wire-frame image and Figure 5 shows the planar contours filled with color.

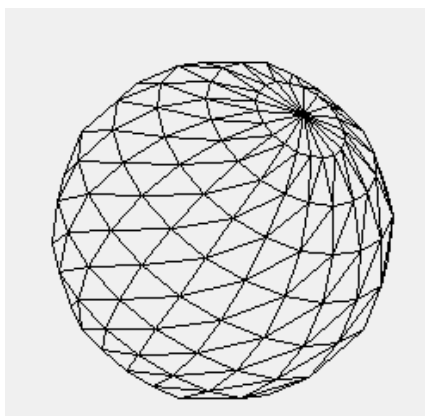
Using the IC model, hidden surface elimination is automatically performed by generating contours in depth order (Painter's algorithm). The illumination model can be minimized to deciding on a range of color values and evaluating a contour's color according to its distance from the light source. For optimum realism, a complex illumination model can be applied. This would achieve results indistinguishable to those produced using ray casting. Ray casting is considered to give the highest degree of realism of all the traditional rendering techniques available. It generates images pixel by pixel and is renowned for the vast amounts of CPU time it consumes. However, the Isoluminance Contour Model produces identical results, give or take a pixel, yet can produce images in a fraction of the time taken to produce even wire-frame.

Comparison of Image Generation Speeds

Table 1 compares the CPU cycles taken to perform a single image containing a thousand spheres using traditional rendering methods on polyhedra with equivalent images produced using the IC model. For this comparison, each sphere was defined by 1250 patches—25 circular planes defined by 50 vertices, and the image was generated on a slow Pentium 400 PC. Although different hardware configurations will yield different results, the relative proportions of these times will remain constant.

For wire-frame drawings, all the edges for the polyhedral representation were drawn, displaying mainly quad-

Figure 3. Flat-shaded sphere represented by triangulated patches



5 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/isoluminance-contours-animated-visualization/14496

Related Content

Challenges of Interoperability in an Ecosystem

Barbara Flüggeand Alexander Schmidt (2009). *Encyclopedia of Information Science and Technology, Second Edition* (pp. 512-518).

www.irma-international.org/chapter/challenges-interoperability-ecosystem/13622

Implementation of E-Learning Functions With the Use of Information Systems Architecture

Mohammed Khaleel Hussein, Raffat Izzat Saheeland Ali Jumaah Ali (2021). *Journal of Cases on Information Technology* (pp. 12-25).

www.irma-international.org/article/implementation-of-e-learning-functions-with-the-use-of-information-systems-architecture/281213

Neural Networks for Retail Sales Forecasting

G. Peter Zhang (2009). *Encyclopedia of Information Science and Technology, Second Edition* (pp. 2806-2810).

www.irma-international.org/chapter/neural-networks-retail-sales-forecasting/13986

An Event-Based Data Warehouse to Support Decisions in Multi-Channel, Multi-Service Contact Centers

Andrea Brunello, Paolo Gallo, Enrico Marzano, Angelo Montanariand Nicola Vitacolonna (2019). *Journal of Cases on Information Technology* (pp. 33-51).

www.irma-international.org/article/an-event-based-data-warehouse-to-support-decisions-in-multi-channel-multi-service-contact-centers/216951

Investigating Web 2.0 Application Impacts on Knowledge Workers' Decisions and Performance

Haya Ajjan, Richard Hartshorneand Scott Buechler (2012). *Information Resources Management Journal* (pp. 65-83).

www.irma-international.org/article/investigating-web-application-impacts-knowledge/70600