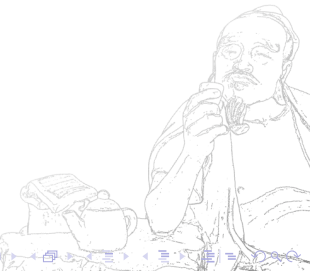
Illustrative Visualization: Photic Extremum Lines

Markus Pawellek

January 10, 2022



Outline

Related Work

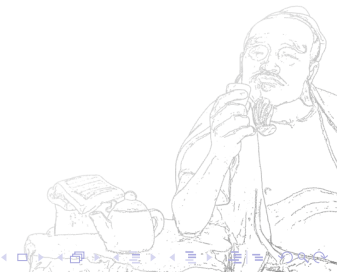
Related Work

Tools

Related Work

Tools

2003 Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models"

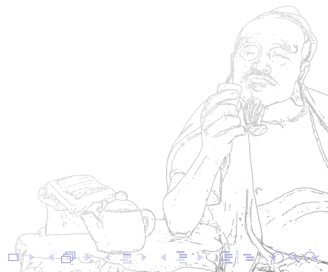


Related Work

Tools

2003 Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models"

2004 Rusinkiewicz "Estimating Curvatures and Their Derivatives on Triangle Meshes"



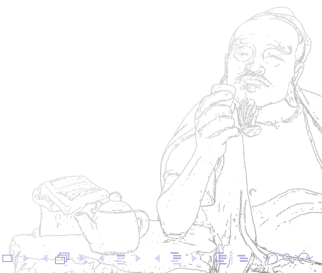
Related Work

Tools

2003 Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models"

2004 Rusinkiewicz "Estimating Curvatures and Their Derivatives on Triangle Meshes"

Algorithm



Related Work

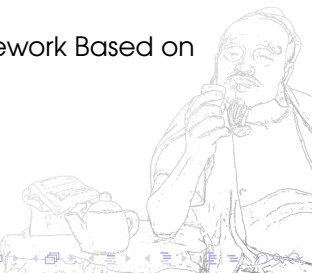
Tools

2003 Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models"

2004 Rusinkiewicz "Estimating Curvatures and Their Derivatives on Triangle Meshes"

Algorithm

2007 Xie et al. "An Effective Illustrative Visualization Framework Based on Photic Extremum Lines (PELs)"



Related Work

Tools

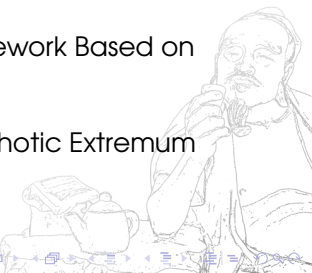
2003 Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models"

2004 Rusinkiewicz "Estimating Curvatures and Their Derivatives on Triangle Meshes"

Algorithm

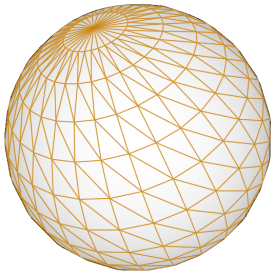
2007 Xie et al. "An Effective Illustrative Visualization Framework Based on Photic Extremum Lines (PELs)"

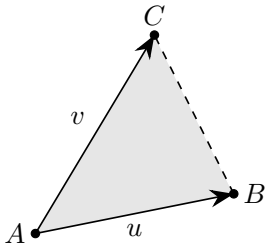
2010 Zhang, He, and Seah "Real-Time Computation of Photic Extremum Lines (PELs)"

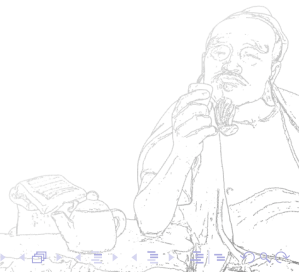


Mathematical Preliminaries

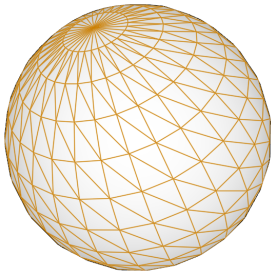
Mathematical Preliminaries: Mesh Function

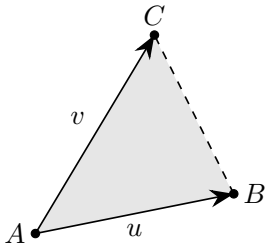






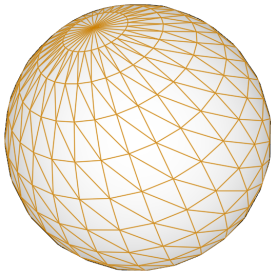
Mathematical Preliminaries: Mesh Function

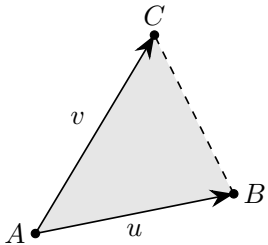




- $f : S \to \mathbb{R}$ on mesh S characterized by its values at vertices

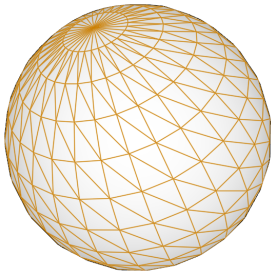
Mathematical Preliminaries: Mesh Function

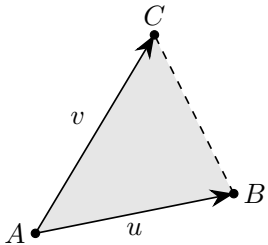


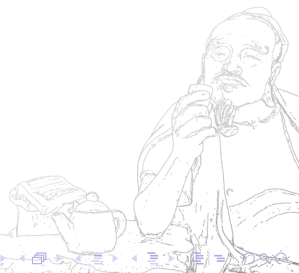


- $f : S \to \mathbb{R}$ on mesh S characterized by its values at vertices
- For interiors of faces, use barycentric interpolation

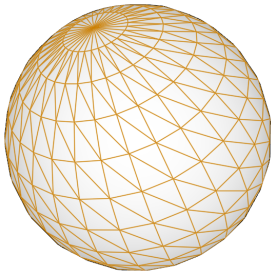
Mathematical Preliminaries: Mesh Function Gradient

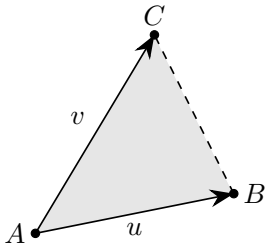




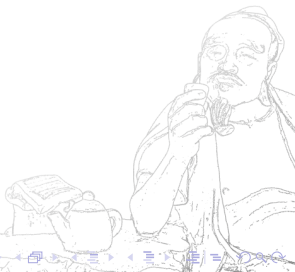


Mathematical Preliminaries: Mesh Function Gradient

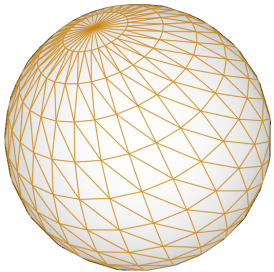


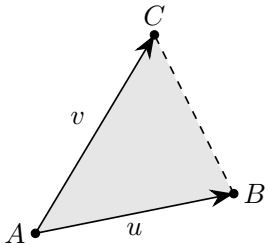


- Compute ∇f for each face



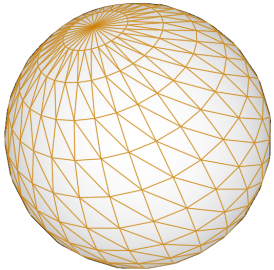
Mathematical Preliminaries: Mesh Function Gradient

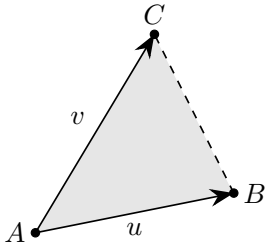




- Compute ∇f for each face
- For each vertex, accumulate weighted gradients for adjacent faces

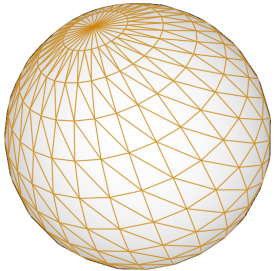
Mathematical Preliminaries: Mesh Function Gradient

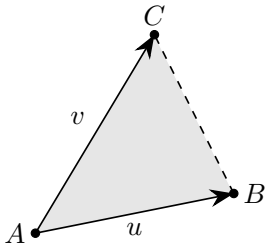




$$\mathrm{I}_{uv} \coloneqq \begin{pmatrix} \|u\|^2 & \langle u, v\rangle \\ \langle u, v\rangle & \|v\|^2 \end{pmatrix} \qquad \nabla f = \begin{pmatrix} u & v \end{pmatrix} \mathrm{I}_{uv}^{-1} \begin{pmatrix} f(B) - f(A) \\ f(C) - f(A) \end{pmatrix}$$

Mathematical Preliminaries: Directional Derivatives



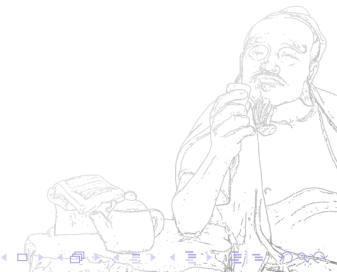


$$\partial_w f(x) = \langle \nabla f(x), w \rangle \qquad \mathcal{D}_f\, g(x) \coloneqq \left\langle \nabla g(x), \frac{\nabla f(x)}{\|\nabla f(x)\|} \right\rangle$$

Photic Extremum Lines

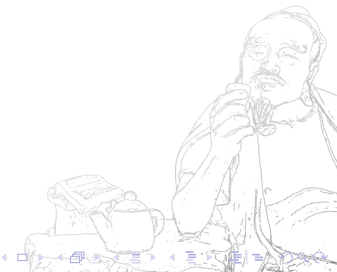
Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)



Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$



Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$

Photic Extremum

Illumination variation in the direction its gradient reaches local maximum

Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$

Photic Extremum

Illumination variation in the direction its gradient reaches local maximum

$$\mathcal{D}_{\varphi}\|\nabla\varphi\|(x) = 0 \qquad \mathcal{D}^2_{\varphi}\|\nabla\varphi\|(x) < 0$$

Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$

Photic Extremum

Illumination variation in the direction its gradient reaches local maximum

$$\mathcal{D}_{\varphi}\|\nabla\varphi\|(x) = 0 \qquad \mathcal{D}^2_{\varphi}\|\nabla\varphi\|(x) < 0$$

Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$

Photic Extremum

Illumination variation in the direction its gradient reaches local maximum

$$\mathcal{D}_{\varphi}\|\nabla\varphi\|(x) = 0 \qquad \mathcal{D}^2_{\varphi}\|\nabla\varphi\|(x) < 0$$

Photic Extremum Lines: Definition

- Scalar illumination function
  $\varphi\colon S \to \mathbb{R}$ on mesh S
  (e.g. directional light source)
- Variation of illumination $\|\nabla\varphi\|$

Photic Extremum

Illumination variation in the direction its gradient reaches local maximum

$$\mathcal{D}_{\varphi}\|\nabla\varphi\|(x) = 0 \qquad \mathcal{D}^2_{\varphi}\|\nabla\varphi\|(x) < 0$$

Algorithm

Algorithm: Overview

1. Compute φ

Algorithm: Overview

1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$

Algorithm: Overview

1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$
3. Compute $\mathcal{D}_\varphi \|\nabla\varphi\|$

Algorithm: Overview

1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$
3. Compute $\mathcal{D}_\varphi \|\nabla\varphi\|$
4. Compute $\mathcal{D}^2_\varphi \|\nabla\varphi\|$

Algorithm: Overview

1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$
3. Compute $\mathcal{D}_{\varphi}\|\nabla\varphi\|$
4. Compute $\mathcal{D}^2_{\varphi}\|\nabla\varphi\|$
5. Detect line vertices on edges by testing for photic extremums

Algorithm: Overview

1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$
3. Compute $\mathcal{D}_{\varphi}\|\nabla\varphi\|$
4. Compute $\mathcal{D}^2_{\varphi}\|\nabla\varphi\|$
5. Detect line vertices on edges by testing for photic extremums
6. Trace and filter out lines by using a threshold

Algorithm: Overview

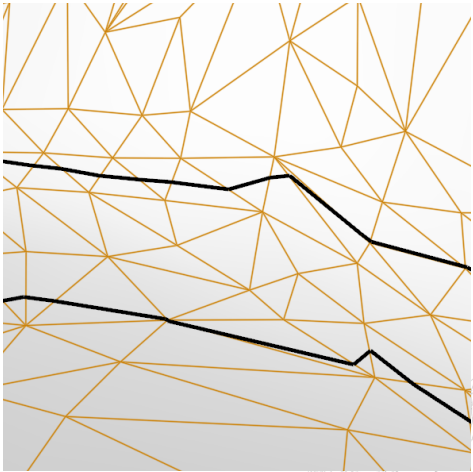
1. Compute φ
2. Compute $\frac{\nabla\varphi}{\|\nabla\varphi\|}$ and $\|\nabla\varphi\|$
3. Compute $\mathcal{D}_\varphi \|\nabla\varphi\|$
4. Compute $\mathcal{D}^2_\varphi \|\nabla\varphi\|$
5. Detect line vertices on edges by testing for photic extremums
6. Trace and filter out lines by using a threshold
7. Render visible lines

Algorithm: Line Detection and Tracing

- For each edge $[v, w] \subset S$, check zero-crossing:

$$h(x) := \mathcal{D}_{\varphi} \|\nabla \varphi\| (x)$$

$$h(v)h(w) < 0$$



Algorithm: Line Detection and Tracing

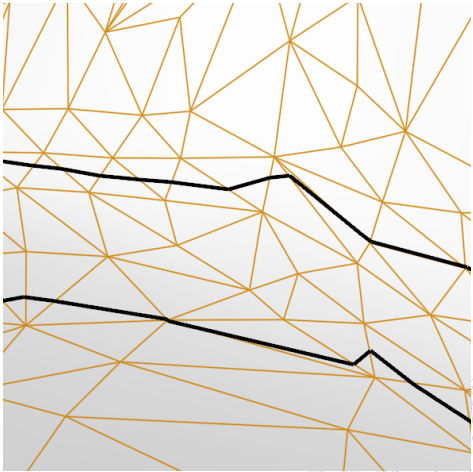
- For each edge $[v, w] \subset S$, check zero-crossing:

$$h(x) := \mathcal{D}_\varphi \left\| \nabla \varphi \right\| (x)$$

$$h(v)h(w) < 0$$

- Approximate zero-crossing:

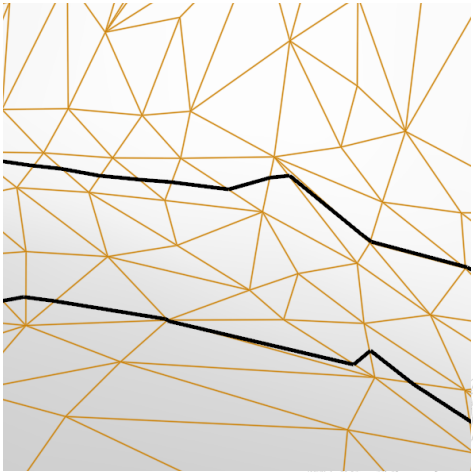
$$p := \frac{|h(w)|\, v + |h(v)|\, w}{|h(v)| + |h(w)|}$$



Algorithm: Line Detection and Tracing

- Check maximum condition:

$$\mathcal{D}^2_{\varphi} \|\nabla\varphi\| (p) < 0$$

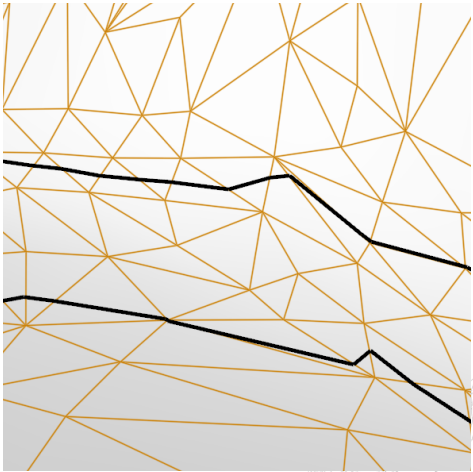


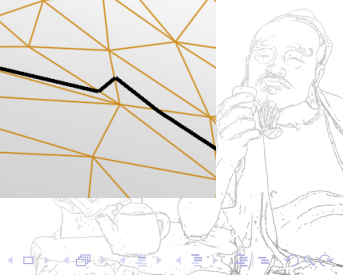
Algorithm: Line Detection and Tracing

- Check maximum condition:

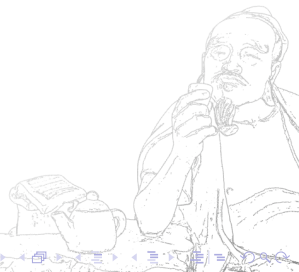
$$\mathcal{D}^2_{\varphi} \|\nabla\varphi\| (p) < 0$$

- For each triangle, connect valid zero-crossings of adjacent edges to segments





Algorithm: Threshold Filter



Algorithm: Threshold Filter

Strength S of photic extremum or strength $\mathcal{S}$ of photic extremum line L:

$$S(x) = \|\nabla\varphi(x)\| > T \qquad \text{or} \qquad \mathcal{S}(L) \coloneqq \int_L \|\nabla\varphi(s)\| \, \mathrm{d}s > T$$

Results

Photic Extremum Lines: Properties

- Object-space method
- View- and light-dependent
- Third- and fourth-order derivatives
-

Problems

preprocessing and good meshes

Conclusions

Thank you for Your Attention!

References

(1) Tobias Isenberg et al. "A Developer's Guide to Silhouette Algorithms for Polygonal Models". In: *Computer Graphics and Applications, IEEE* 23 (August 2003), pp. 28–37. DOI: 10.1109/MCG.2003.1210862.

(2) Szymon Rusinkiewicz. "Estimating Curvatures and Their Derivatives on Triangle Meshes". In: October 2004, pp. 486–493. ISBN: 0-7695-2223-8. DOI: 10.1109/TDPVT.2004.1335277.

(3) Xuexiang Xie et al. "An Effective Illustrative Visualization Framework Based on Photic Extremum Lines (PELs)". In: *IEEE transactions on visualization and computer graphics* 13 (November 2007), pp. 1328–1335. DOI: 10.1109/TVCG.2007.70538.

(4) Long Zhang, Ying He, and Hock Seah. "Real-Time Computation of Photic Extremum Lines (PELs)". In: *The Visual Computer* 26 (June 2010), pp. 399–407. DOI: 10.1007/s00371-010-0454-x.

(5) Douglas DeCarlo et al. "Suggestive Contours for Conveying Shape". In: *ACM Trans. Graph.* 22 (July 2003), pp. 848–855. DOI: 10.1145/1201775.882354.

(6) Nelson Max. "Weights for Computing Vertex Normals

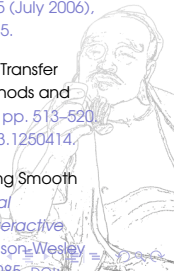
(8) Long Zhang et al. "Real-Time Shape Illustration Using Laplacian Lines". In: *IEEE transactions on Visualization and Computer Graphics* 17 (July 2011). DOI: 10.1109/TVCG.2010.118.

(9) Michael Kolomenkin, Ilan Shimshoni, and Ayellet Tal. "Demarcating Curves for Shape Illustration". In: *ACM Trans. Graph.* 27 (December 2008), p. 157. DOI: 10.1145/1457515.1409110.

(10) Szymon Rusinkiewicz, Michael Burns, and Douglas DeCarlo. "Exaggerated Shading for Depicting Shape and Detail". In: *ACM Trans. Graph.* 25 (July 2006), pp. 1199–1205. DOI: 10.1145/1179352.1142015.

(11) Gordon Kindlmann et al. "Curvature-Based Transfer Functions for Direct Volume Rendering: Methods and Applications". In: vol. 2003. November 2003, pp. 513–520. ISBN: 0-7803-8120-3. DOI: 10.1109/VISUAL.2003.1250414.

(12) Aaron Hertzmann and Denis Zorin. "Illustrating Smooth Surfaces". In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. ACM Press/Addison-Wesley



Previous Work