A Fast Method to Calculate the Volumetric Divergence Metric for Evaluating the Accuracy of the Extracted Isosurface

Cuilan Wang
Georgia Gwinnett College

High-Performance Graphics 2018
August 10-12, Vancouver, BC
Isosurface Extraction

• Usually: 3D scalar rectilinear volumes

• Trilinear interpolation
  • Commonly used to model the interior of each dataset cell especially when the underlying data function is unknown

\[
F(x, y, z) = F_{000}(1 - x)(1 - y)(1 - z) \\
+ F_{100}x(1 - y)(1 - z) + F_{010}(1 - x)y(1 - z) \\
+ F_{110}xy(1 - z) + F_{001}(1 - x)(1 - y)z \\
+ F_{101}x(1 - y)z + F_{011}(1 - x)yz + F_{111}xyz,
\]

where \( F_{ijk} \) denotes the value at cube corner, \( i,j,k = 0,1, \) and \((x, y, z)\) are local coordinates in a cube, \( 0 \leq x \leq 1, 0 \leq y \leq 1, 0 \leq z \leq 1.\)
Isosurface Extraction Algorithms

• Example: Marching Cubes

• Produce a triangular mesh to approximate the isosurface given by trilinear interpolation

• Accuracy
  • Closeness of the mesh and the trilinear interpolation isosurface
Estimating Accuracy of the Produced Mesh

• Benefits
  • Researchers: design new isosurfacing algorithms
  • The extracted mesh with high accuracy are useful:
    • Users need to substantially zoom into the data.
    • Medical applications where thin specimens or internal small cavities need to be viewed.
Volumetric Divergence (VD) Metric

• Volumetric divergence of two closed surfaces A and B:
  \[ V(R) \text{, where } R = (A_{\text{in}} \cap B_{\text{out}}) \cup (A_{\text{out}} \cap B_{\text{in}}). \]

• It is a natural and accurate metric to measure the closeness between two surfaces.

• To measure the accuracy of the extracted isosurface
  A: extracted triangular mesh
  B: trilinear interpolation isosurface
Volumetric Divergence Calculation

• The previous method (i.e., VDC method) [1] is a numerical approach
  1. Recursively subdivide a cube into 8 subcubes until a certain subdivision depth is reached or other termination criteria are satisfied.
  2. Determine each subcube’s contribution to the volumetric divergence and sum the contributions.

• When the subdivision depth is high, the produced VD metric is very accurate, but the execution time is very long
Goal

• Develop a new VD calculation method faster than the previous method when both methods achieve the same level of accuracy
  • Make the VD metric used in real time to benefit isosurface extraction algorithms, such as Multi-Resolution MC [2]

• Solution
  – Combine an analytical approach and a numerical approach
Related Work
Measure Representational Accuracy

- Global geometric metrics
  - Surface area, volume inside the meshes [3]
  - Do not measure local deviation well

- Distance based metrics
  - Quadric error metric [4], Hausdorff distance [5]
    - Consider only some points (e.g., mesh vertices) of the mesh
  - Metro Scheme [6]
    - At densely sampled locations

- Shortcomings
  - Not exhaustive
  - One-sided (i.e., not symmetric)
  - Approximate measures of mesh distance
The Slicing Method
The Overall Method

• Analytical approach
  • On each slice, the cross section of the VD region is enclosed by the isocontour hyperbola and the iso-polyline. Its area can be calculated analytically.

• Numerical approach
  • Similar to integrating areas to find volume
The Overall Method

• Suppose the cube is evenly sliced along the $y$-axis. The volume of the VD region is approximated by

$$\sum_{i=1}^{n} A(y_i) \Delta d$$

, where $n$ is the number of slices, $\Delta d = 1/n$, and $A(y_i)$ is the cross-sectional area of the VD region on the slice $y = y_i$.

• The more that the cube is sliced, the more accurate the result is.
Area Under Hyperbola

• The bilinear interpolation function over a slice is

\[ B(x, z) = B_{00}(1 - x)(1 - z) + B_{10}x(1 - z) + B_{01}(1 - x)z + B_{11}xz \]

• It can be rewritten as

\[ B(x, z) = axz + bx + cz + d, \]

where \( a = B_{00} - B_{10} - B_{01} + B_{11}, b = B_{10} - B_{00}, c = B_{01} - B_{00}, \) and \( d = B_{00}. \)
Area Under Hyperbola

• Let $\beta$ be the isovalue.

• The bilinear interpolation isocontour is defined as

$$C_B \equiv \{ (x, z) : B(x, z) = \beta \},$$

• It can be rewritten as

$$(x - u)(z - v) = w,$$

where $u = -\frac{c}{a}$, $v = -\frac{b}{a}$, and $w = \frac{bc - ad + a\beta}{a^2}$. 

• It can be expressed as $z$ being a function of $x$:

$$z = f(x) = \frac{-bx - d + \beta}{ax + c}.$$
Area Under Hyperbola

- Integrating $z = f(x)$ between the limits of $x_1$ and $x_2$ to get the area under hyperbola:

\[
\text{area} = \int_{x_1}^{x_2} f(x)\,dx = v(x_2 - x_1) + w \ln \left| \frac{x_2 - u}{x_1 - u} \right|.
\]
Slice Partition

- The iso-polyline may intersect with the isocontour hyperbola
Slice Partition

- To partition, use the x values of the following points:
  - Iso-polyline vertices
  - Intersection points of isocontour hyperbola and slice edges
  - Intersection points of iso-polylines and isocontour hyperbola
Subregion Division

- The iso-polylines and the isocontour hyperbola divide a subregion into several segments.
- The segments that are inside one type of the surface but outside the other belong to the VD region.
Calculating the Area of a Subregion Segment

- Area under the upper boundary (b) subtracting the area under the lower boundary (c) yields the subregion segment area (a).
Results
Experimental Settings

• The unit of the VD metric is the volume of a unit cube, which is set to 1.

• Two depths
  • VDC method – subdivision depth \( s \)
  • Slicing method – partition depth \( p \)
    • Number of slices = \( 2^p \)
Individual Cube Instances

- Three instances of base case 6.1.1, 6.1.2, and 6.2

6.1.1

6.1.2

6.2

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Extracted Isosurface

Tri. Inter. Isosurface

overlay
Individual Cube Instances

- Both methods converge and they converge to the same set of values.
Randomly Generated Instances

• Test 1000 randomly generated instances for each of three isovalues: 63, 127, and 191
• Use the VDC method with subdivision depth 10 as the baseline
## Randomly Generated Instances

### The Slicing Method

<table>
<thead>
<tr>
<th>Partition Depth</th>
<th>ISOvalue = 63</th>
<th>ISOvalue = 127</th>
<th>ISOvalue = 191</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.001995</td>
<td>0.021705</td>
<td>0.003</td>
</tr>
<tr>
<td>4</td>
<td>0.000953</td>
<td>0.009922</td>
<td>0.006</td>
</tr>
<tr>
<td>5</td>
<td>0.000468</td>
<td>0.005099</td>
<td>0.012</td>
</tr>
<tr>
<td>6</td>
<td>0.000233</td>
<td>0.002569</td>
<td>0.022</td>
</tr>
<tr>
<td>7</td>
<td>0.000116</td>
<td>0.001295</td>
<td>0.047</td>
</tr>
<tr>
<td>8</td>
<td>0.000058</td>
<td>0.000647</td>
<td>0.078</td>
</tr>
<tr>
<td>9</td>
<td>0.000029</td>
<td>0.000323</td>
<td>0.171</td>
</tr>
<tr>
<td>10</td>
<td>0.000014</td>
<td>0.000162</td>
<td>0.344</td>
</tr>
<tr>
<td>11</td>
<td>0.000007</td>
<td>0.000081</td>
<td>0.670</td>
</tr>
<tr>
<td>12</td>
<td>0.000004</td>
<td>0.000041</td>
<td>1.326</td>
</tr>
<tr>
<td>13</td>
<td>0.000002</td>
<td>0.000021</td>
<td>2.652</td>
</tr>
</tbody>
</table>
## Randomly Generated Instances

### The VDC Method

<table>
<thead>
<tr>
<th>Subdivision Depth</th>
<th>Isovalue = 63</th>
<th>Isovalue = 127</th>
<th>Isovalue = 191</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.002245</td>
<td>0.025474</td>
<td>0.090</td>
</tr>
<tr>
<td>4</td>
<td>0.000602</td>
<td>0.007285</td>
<td>0.283</td>
</tr>
<tr>
<td>5</td>
<td>0.000156</td>
<td>0.002632</td>
<td>0.961</td>
</tr>
<tr>
<td>6</td>
<td>0.000039</td>
<td>0.000562</td>
<td>3.291</td>
</tr>
<tr>
<td>7</td>
<td>0.000010</td>
<td>0.000143</td>
<td>11.801</td>
</tr>
<tr>
<td>8</td>
<td>0.000002</td>
<td>0.000038</td>
<td>43.817</td>
</tr>
</tbody>
</table>
Randomly Generated Instances

- Map the VDC method and the Slicing method with the same level of accuracy.

<table>
<thead>
<tr>
<th>Isovalue = 63</th>
<th>Isovalue = 127</th>
<th>Isovalue = 191</th>
</tr>
</thead>
<tbody>
<tr>
<td>VDC</td>
<td>Slicing</td>
<td>VDC</td>
</tr>
<tr>
<td>3</td>
<td>0.002245</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>0.000602</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>0.000156</td>
<td>7</td>
</tr>
<tr>
<td>6</td>
<td>0.000039</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>0.000010</td>
<td>11</td>
</tr>
<tr>
<td>8</td>
<td>0.000002</td>
<td>13</td>
</tr>
</tbody>
</table>
Randomly Generated Instances

- Performance comparison

<table>
<thead>
<tr>
<th>Isovalue</th>
<th>VDC Subd. Depth</th>
<th>VDC Time (sec.)</th>
<th>Slicing Partion Depth</th>
<th>Slicing Time (sec.)</th>
<th>Speedup Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>63</td>
<td>3</td>
<td>0.090</td>
<td>3</td>
<td>0.003</td>
<td>30.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.283</td>
<td>5</td>
<td>0.012</td>
<td>23.6</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.961</td>
<td>7</td>
<td>0.047</td>
<td>20.4</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3.291</td>
<td>9</td>
<td>0.171</td>
<td>19.2</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>11.801</td>
<td>11</td>
<td>0.670</td>
<td>17.6</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>43.817</td>
<td>13</td>
<td>2.652</td>
<td>16.5</td>
</tr>
<tr>
<td>127</td>
<td>3</td>
<td>0.220</td>
<td>3</td>
<td>0.009</td>
<td>24.4</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.805</td>
<td>5</td>
<td>0.031</td>
<td>26.0</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>2.692</td>
<td>7</td>
<td>0.109</td>
<td>24.7</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>9.355</td>
<td>9</td>
<td>0.452</td>
<td>20.7</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>33.762</td>
<td>11</td>
<td>1.794</td>
<td>18.8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>125.556</td>
<td>13</td>
<td>7.191</td>
<td>17.5</td>
</tr>
<tr>
<td>191</td>
<td>3</td>
<td>0.089</td>
<td>3</td>
<td>0.003</td>
<td>29.7</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.295</td>
<td>5</td>
<td>0.013</td>
<td>22.7</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1.015</td>
<td>7</td>
<td>0.047</td>
<td>21.6</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3.475</td>
<td>9</td>
<td>0.172</td>
<td>20.2</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>12.307</td>
<td>11</td>
<td>0.734</td>
<td>16.8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>45.755</td>
<td>13</td>
<td>2.932</td>
<td>15.6</td>
</tr>
</tbody>
</table>
Real Datasets

- Performance comparison

<table>
<thead>
<tr>
<th>VDC</th>
<th>Slicing</th>
<th>Speedup Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.002581</td>
<td>10.595</td>
</tr>
<tr>
<td>4</td>
<td>0.000715</td>
<td>36.808</td>
</tr>
<tr>
<td>5</td>
<td>0.000183</td>
<td>124.792</td>
</tr>
<tr>
<td>6</td>
<td>0.000043</td>
<td>430.097</td>
</tr>
<tr>
<td>VDC</td>
<td>Slicing</td>
<td>Speedup Factor</td>
</tr>
<tr>
<td>3</td>
<td>0.002768</td>
<td>16.868</td>
</tr>
<tr>
<td>4</td>
<td>0.000756</td>
<td>58.346</td>
</tr>
<tr>
<td>5</td>
<td>0.000191</td>
<td>195.977</td>
</tr>
<tr>
<td>6</td>
<td>0.000045</td>
<td>675.075</td>
</tr>
</tbody>
</table>

Isovalue = 45.5
- # of active cubes = 114371, avg. VD = 0.023200
- MC time = 0.20 sec.

Isovalue = 85.5
- # of active cubes = 164229, avg. VD = 0.029655
- MC time = 0.27 sec.

(a) isovalue = 45.5
(b) isovalue = 85.5
Real Datasets

- Rendering of two real datasets where the mesh has large error is rendered in color
Application of the Slicing Method in Adaptive Isosurface Extraction

• Three methods are compared
  • Marching Cubes (MC)
  • Multi-Resolution Marching Cubes (MRMC) in [2]
    • Use the Slicing method to calculate the VD metric values
    • Mesh at the cubes where the VD metric values are larger than a threshold is refined.
  • Dividing Cubes with subdivision depth 1 (DC1)
Application of the Slicing Method in Adaptive Isosurface Extraction
Conclusion and Future Work
Conclusion

• Both methods are correct and are capable of producing highly accurate VD metric values.
• The new method is about 15-70 times faster than the previous method.
Potential Future Work

• Modify the method to compute
  • Volume between two meshes
    • Check the closeness of the two meshes
  • Volume between two isosurfaces
    • Reveal the change rate of the isosurfaces
      • May be used to guide the selection of the representative isovalues of a dataset
  • Volume between a mesh and an isosurface where the underlying data function is known, but not trilinear interpolation isosurface
Questions?
Thank you!
References


