Volume Visualization using Marching Cubes Algorithms: Survey & Analysis

Bhumika N. Parmar[1], Mr. Tejas Bhatt[2]
[1]*M.E. Student, Dept. of C.E., S.V.I.T. Vasad, Gujarat, India*
[2]*Assistant Professor, Dept. of C.E., S.V.I.T. Vasad, Gujarat, India*

***Abstract*— This paper presents a study of the algorithm used for generating 3D structures from 2D datasets. This is achieved by developing an execution of Marching Cubes, surface rendering algorithm that's at present the typical used for 3D surface structure in the medical visualization industry. Three Dimensional (3D) image visualization is one of the significant processes that extract in order from a known single image. This paper contains survey on marching cube algorithms and extension to it. The Marching Cubes (MC) algorithm by Lorensen and Cline is most used algorithm for extraction of isosurface out of volumetric data. A number of drawbacks of MC algorithm are solved by using new enhanced version of MC algorithm.**

***Index Terms*— isovalue, isosurface extraction, Marching cubes, Volume rendering, Volumetric data.**

I. INTRODUCTION

Volume visualization is a technique of take-out meaningful information from volumetric data. Volume visualization has been useful in lots of difficulty domains and, has turn into a significant implement for exploring data and discovering information. Volume visualization is used to produce images from scalar and vector datasets distinct on multiple dimensional grids. It is a new but quickly rising field in both data visualization and computer graphics. These techniques are used in geoscience, medicine, chemistry, microscopy, astrophysics, mechanical engineering, and other areas.[9]

Volume visualization is a set of techniques used to display a 3D projection of a 2D data set, naturally a 2D cross-sectional images. A standard 3D data set is a combination of 2D slice images acquired by a MRI, CT, or Micro CT scanner. Generally these are acquired in a normal pattern (e.g., one slice apart every mm) and generally have a constant number of image pixels in a regular pattern. This is an ex. of a regular volumetric grid, with each voxel represented by a single cube that is gain by sampling the instant region nearby the voxel.

Human brain is the most vital and compound organ of our central nervous system. It is really important to convert available 2D MRI images to a 3D brain image to determine if there are any irregularities. 3-D rendering of medical images is broadly applied to tumor localization, medical planning and eliminating the need for rescanning.[13] The brain MRI images contain of wide range of the gray-scales and highly unequal boundaries. So it is problematic to categorize different matters using present approaches. Earlier, the understanding of 3D imagining was mainly by two renovation approaches, one is Indirect Volume Rendering (surface rendering) and the other is Direct Volume Rendering (volume rendering).

This paper will first outline the approaches of image rendering. Then the standard Marching Cube algorithm is defined. Then different adjustment in Marching Cube algorithm for more effective and fast rendering are defined. Next is evaluation criteria of the algorithm is explained. Then the disadvantages of Marching Cubes are defined.

II. METHODS

2.1 Image Rendering:

Rendering is a computer graphics development for imagining 2D images with 3D properties without shifting hardware. 3D visualization was mainly executed by two reconstruction methods .[9]

- **Direct Volume Rendering:** The DVR is defined as direct element mapping onto the screen space, without the use of geometric primitives as an intermediary representation.[9] In DVR transparency and colors are used for the better 3-D visualization of the volume. For ex. Pelvis bones can be displayed as semi-transparent, So that at some angle, one part of the image does not cover up another.

- **Indirect Volume Rendering:** The IVR also called the surface fitting (SF) subsist of stages of feature extraction and *isosurfaces*, which is rendered for the visualization. Example of SF is Marching Cubes and contour connection.[9]

2.2 The Concept of "*iso*":

An "*isosurface"* can be defined as a set of points that have the same value i.e. "*isovalue"*.[9] The "*isosurface"* extraction generally involves production of an approximate "*isosurface*" on a sampled scalar field.[6] The generation of isosurface procedure involves the creation of meshes (generally triangular) that approximately represents a certain surface.

The rendered image surface depends on value of 'ISO' as shown in fig.1 (a). Actual outlines would be produced from voxels by use of cubes as shown in fig. 1(b).

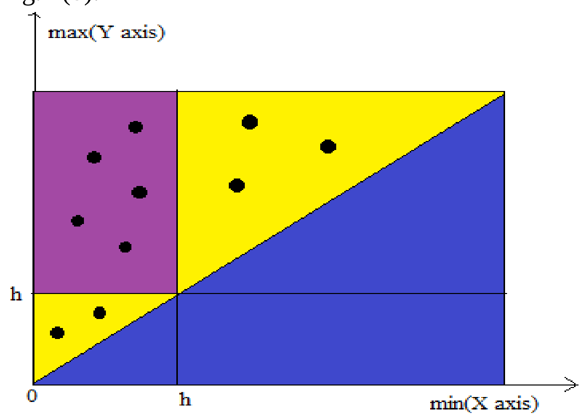


Fig. 1(a) Impact of *ISO* selection

In Fig. 1(a), the lowest (V_{min}) and highest (V_{max}) vertex value are signified by an opinion in span space. Given an ISO value, only cells that content both ($V_{min} \leq V_{iso}$ and $V_{max} \geq V_{iso}$) and contain the ISO surface are more treated.

2.3 The concept of voxels:

Voxel is the basic unit in 3-dimensional where it can be developed by four points in each parallel slices into a cube. On each side of the orthogonal axes, voxel has the shape of a hexahedron. Voxel uses the characterized points to coordinate the direction of constantly increasing form, as shown in Figure 1(b).

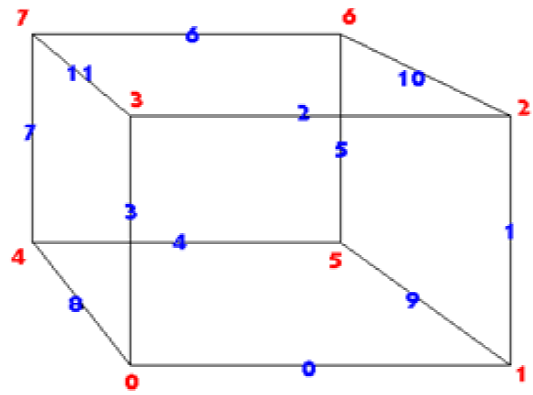


Fig. 1(b) Vertices Generation

III. THE STANDARD MARCHING CUBES ALGORITHM

Marching Cubes algorithm is the most widely used Surface Rendering algorithm for the *isosurface* extraction. *Marching cubes* uses a divide-and-conquer approach in which volume data is processed through voxels..[1] For the processing, multi-image slices are arranged as a multidimensional array. And two adjacent slices are taken into consideration at a time as shown in figure 2.

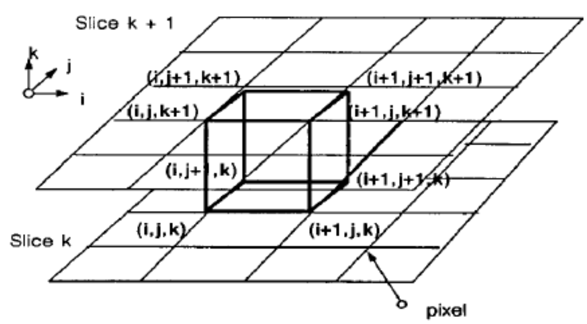
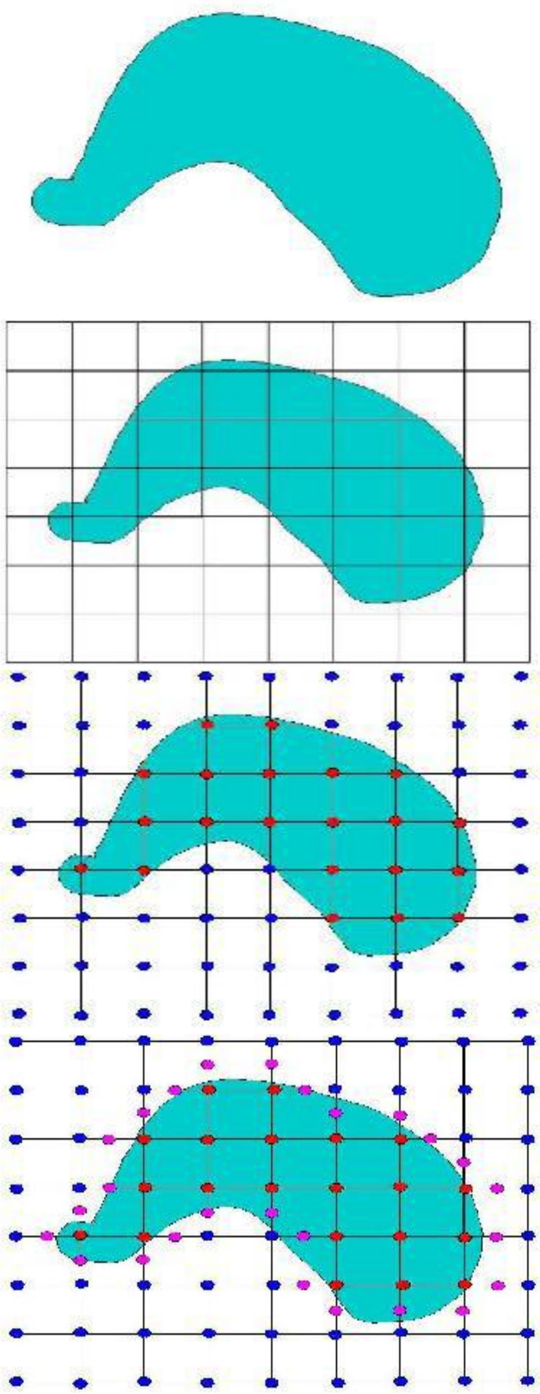


Fig 2. Cube formed on lattice.points[1]

Divide the interplanetary inside the limits into a random number of cubes. Then after, the intersection between edges and the isosurface is verified. Test the vertices of every cube for whether they are inside the object. For each cube where some vertices are internal and some vertices are external the object, the surface must pass over that cube, crossing the edges of the cube in between corners of conflicting organization. When the value of the vertex is greater than or equal to isovalue it is internal, and when it is less than isovalue it is external.[1] You have your object shown in fig 3(a).



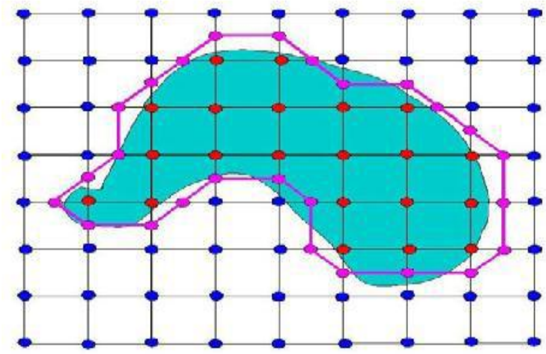


Fig. 3. Pre-Processing for standard Marching Cube. (A)Image Object (B) Segmented Image (C) Intersected Point Finding (D) Find Offset Points (E) Join all Offset points.

Each 8 vertex cells has only two possible states, there is total of $2^8 = 256$ cases of intersection between isosurface and edges. For the faster execution they are listed in lookup table. We can reduce the 256 cases into 15 unique marching cubes, which is shown in figure 4.

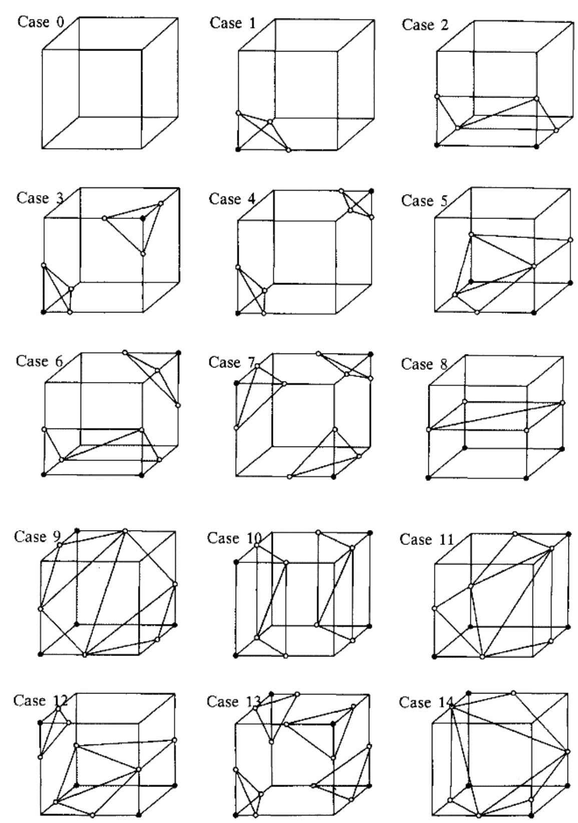


Fig 4. Unique 15 Cubes

IV. MARCHING CUBES EXTENSIONS

The Marching Cube has been extended in following ways.
(1) Computational improvements that limit needless effort, especially during traversal, or that use parallel and dispersed processing.
(2) Hurtling up the marching cubes algorithm on a graphics dispensation unit.
(3) Removal of Ambiguity

A brief Survey:

Lei Guo, Ming Hu, YingLi, Weili Yan[10] presented that the standard MC visits all cubes including active and non-active cubes by sequential traversal in the process of the isosurface extraction from volumetric data sets, which is inefficient and time consuming . In this study, the standard MC algorithm is combined with the seeded region growing algorithm, and an improved MC algorithm is proposed. The new algorithm aim to avoid the computation of non-active cubes. By experiments and theoretical analysis it is proven that this algorithm increases the speed of 3-D regeneration of images and removes the noise from the segmentation stage. Furthermore, mesh smoothing and normal calculation are included to improve the rendering effect.

Newman and Yi[6] industrialized an in-depth study about the potentials of developing the marching cubes procedure, recitation their properties, delays and efforts to solve its boundaries. The improvements, requests, and spin-offs described in the graphics and imagining works have also been labelled. Though, the paper only shows the differences among these potentials in terms of algorithm difficulty as well as imagining results, and does not make an examination with volume datasets.

Marcos Vinicius Mussel $Cirne$[9] defines a procedure for moving up the marching cubes algorithm on a graphics processing unit and deliberates a number of ways to progress its presentation through supplementary three-dimensional data structures. Experiments shown with the use of numerous volumetric datasets validate the efficiency of the industrialized technique. Time difficulty of the algorithm be contingent on the volume data cells that comprise an ISO superficial, in its place of the total number of cells, evading the dispensation of unfilled cells.

Smistad et al.[8] obtainable an application of the marching cubes algorithm written in Open CL that runs completely in the GPU. It uses the histogram pyramids to produce the output stream of vertices to be reduced and runs as fast as CUDA and shared applications, providing a very effective storing arrangement as well. However, the disadvantage of their application lies on the interoperability between Open CL and Open GL.

Challenges:

The size of datasets is in the MB to GB. An aim is that two dissimilar approaches of procurement data, for example, a CT scan and MRI, may give dissimilar information. The operator may poverty to view both data sets, spatially registered so they can understand the features of both. Also the scheme might display the most suitable information from every data set at every point. The version must be fast sufficient to permit for communication, i.e., the user must be intelligent to alteration limits and see the effect quickly.

V. EALUATION CRITERIA

5.1 *isovalue*:

In rendering process, the marching cube algorithm is based on convenient choosing of "*isovalue"*. The visible part or the quality of rendered image depends on "*isovalue"*. It is a kind of threshold value.

5.2 Processing Time:

The rendering processing should be fast enough. We can evaluate it based on the processing time.

VI. DRAWBACKS OF MC ALGORITHM

1. Triangular Patch Constructed by MC is Only Confined to Approximate Expression of object Surface:

In marching cubes algorithm, intersection of object surface and voxel boundary is considered from that belief that intersection point varies according to voxel boundary. Image is with high density and small voxel, this belief is close to actual situation. But in image with low density and large voxel, large error will be resulted. Thus the exact comparison would be figured out, when apply different eligible cases of voxel boundary based on different background.[12]

2. High time difficulty:

For smoother image, we need more vertices. But as number of vertex increase more time required to generate the boundary. Which increases higher time complexity.

3. Degenerate Triangles:

When the data set contains lattice points whose values equal the isovalue, some isosurface triangles may be degenerate (i.e., have collinear vertices). One way to solve the redundancy problem is to choose an *isovalue* that is definite from all scalar values.[6]

VII. CONCLUSION

Rendering of medical data in 3D is an appreciated tool that will help doctors in identifying abnormalities and treating with better sureness. We serving that when we using this algorithms such as Marching Cubes, 3 dimensional images can be produced with relative comfort. We can decrease the processing time in generating result if we integrate this algorithm with graphical processing unit. It was conceivable to speed up the marching cubes algorithm by an issue of around 18 times when associated to the CPU method using GPU. In future we can increase processing speed by integrating numerous shading languages with MATLAB.

REFERENCES

[1] William E. Lorensen, Harvey E. Cline, “Marching Cubes: A high resolution 3D surface construction algorithm” Computer graphics, volume 21, July 1987.

[2] Cline, H. E., Lorensen *Medical Physics*, 15, 3 (May/June 1988), 320-327, W. E., Ludke, S., Crawford, C. R. and Teeter, B. C., ‘‘Two Algorithms for the Three-Dimensional Reconstruction of Tomograms’’.

[3] George M. Nielson, Bernd Hanmann “The asymptotic decider: Resolving the Ambiguity in Marching cubes” IEEE 1991.

[4] T. Lewinera, H. Lopes, A. W. Vieira and G. Tavares, “Efficient implementation of marching cubes’ cases with topological guarantees,” Journal of Graphics Tools, vol. 8, no. 2, pp. 1–15, 2003.

[5] L. Adriano, “Improving the robustness and accuracy of the marching cubes algorithm for isosurfacing,” IEEE Transactions on Visualization and Computer Graphics, vol. 9, no. 1, pp. 16–29, March 2003.

[6] Timothy S. Newman “A survey of the marching cubes algorithm” Elsevier Science Direct (Computer & Graphics (30)); 2006. p. 854-879.

[7] Khamphong KHONGSOMBOON, Kazuhiko HAMAMOTO “Proposal of Differential Volume Rendering Reconstruction from X-ray Fluoroscopy for Clinical Veterinary Medicine” IEEE 2008.

[8] Smistad E, Elster AC, Lindseth F. “Fast surface extraction and visualization of medical images using openCL and GPUs” 2011.

[9] Marcos Vinicius Mussel Cirne “Marching cubes technique for volumetric visualization accelerated with graphics processing units” Springer 2012.

[10] Lei Guo, Ming Hu, YingLi, Weili Yan “Three Dimension Reconstruction of Medical Images Based on an Improved Marching Cubes Algorithm” IEEE 2013

[11] MuzhirShaban Al-Ani, Shokhan Mahmoud H. “3D visualization based on surface estimation techniques” IJAET May 2013

[12] Shuqin Liu, Jinye Peng “Optimization of Reconstruction of 2D Medical Images Based on Computer 3D Reconstruction Technology” JDIM June 2015

[13] G. Li, J. Tian, M. Zhao and H. He, “Three–dimensional interactive virtual endoscopy,” Journal of X-Ray Science and Technology, vol. 12, no. 3, pp. 129–141, March 2004.