Different Ray-Casting Algorithm Implementations for Volume Rendering

Palak Chatwani, Shivani Shah, Purvi Ramanuj

Abstract - Volume Rendering is the way to achieve 3D visualization. Volume Rendering is used for visualization of 2D projections of 3D data. In volume rendering techniques, direct volume rendering techniques (DVR) can be divided into image order and object order. Image order technique can be achieved by ray-casting algorithm. Ray-casting algorithm is used for ray-surface interaction tests to solve problems in computer graphics like collision detection and hidden surface removal. In DVR, the ray is pushed through the object and 3D scalar field of interest is sampled along the ray inside the object. Over the years, different approaches towards this algorithm took place. This paper represents the review and analysis of different approaches of ray-casting algorithm.

Keywords - Volume Rendering, Ray-casting, 3D visualization

I. INTRODUCTION

Volume rendering is the key element in medical imaging (for Computer Tomography (CT), Magnetic Resonance Imaging (MRI)), ODT[14] 3D gaming with graphic cards, multimedia, geosciences, remote sensing, etc. Essentially, volume rendering standards were grown by Marc Levoy firstly and further developed by Robert A. Drebin and some other researchers participated in the field of representation. Levoy gave the point by point review which executed Volume rendering in various fields [1].

Volume rendering works on rendering pipeline. In the rendering pipeline, volumetric information is gathered from different sources like MRI, CT scan, PET, etc. Generally, a volume related group of information is situated as V of the examples (x, y, z, v), likewise named voxels, speaking to the worth v of some attribute of the information, on 3 dimensional area (x, y, z). In the next phase, the interpolation is done. Interpolation can be defined as the generation of new values based on existing values. Linear, bilinear and trilinear interpolation can take place as per the dimensions. Measurement of light which is redirected from volumetric exteriors towards the human eye is assessed by Gradient estimation. After Gradient estimation, classification and Shading is done parallely. Classification can be done using a transfer function. In classification, darkness and shading are mapped with information qualities, that elements can be covered up or stressed with the outcome.

The procedure of shading refers to figure the illumination prototype where the result can be defined as the shading for distinct pixels or a polygon.

Next step is applying the lighting. Lighting models are used to compute the light conduct and specifically how light sources constituent and geometry collaborates. Indirect volume rendering extracts certain areas of interest from the entire volume and transforms them into polygonal models E.g. marching cubes [12], which approximates a polygonal model from a voxel-based dataset. Compositing is the final stage to reach the image by binding all steps of rendering. Compositing can be done using back-to-front or front-to back traversal [1].

II. RELATED WORK

Volume rendering techniques can be categorized into 2 categories. 1) Direct Volume Rendering (DVR) 2) Indirect Volume Rendering (IVR).

Direct volume rendering primarily offers flexibility, it can be used to obtain an initial overall view of the data, and, by changing transfer functions (which are directly analogous to color maps), data’s particular features can also be focused incrementally. Direct volume-rendering algorithms consist of three major components: sampling, classification, and compositing. Sampling deals with selecting the piecewise steps that are taken through the volume; classification is the process of computing a color and opacity for each step using the volume-rendering integral; and compositing blends these classified steps together to form an image as a final step [2].

DVR can be classified as Image Order techniques and Object Order techniques. In the image order technique, pixels get segmented amongst processors. Segmentation generates successive pixels containing groups, which are called tiles. Every processor initiates the process with loading the cells which contribute to their tiles. After that every processor creates a part of an image corresponding to the tile, by executing the cells which are loaded. Part of the image generated by every processor is called a sub image. After that, the sub images from all the processors get assembled on a single processor to create the resulting image [3]. Ray casting is an example of Image order technique.

Where in object order technique, the dataset is segmented in blocks amongst the processors. Every processor renders their own cells from other processors, independently. After that the contributions from all processors are combined together to generate a final image, e.g. splatting [11]. Shear-warpping and splatting are object order techniques.

Splatting technique gets compared with tossing a voxel (voxel looking like a snowball) at a dish made of glass.

Revised Manuscript Received on May 20, 2020.

Ms. Palak Chatwani, Pursuing M.E. at L.D. College of Engineering and doing her dissertation at DWD/MRG/EPSA SAC, ISRO.

Mrs. Shivani Shah, Scientist-SF at DWD/MRG/EPSA at SAC, ISRO. She works as a MOSDAC administrator as well as principal investigator cloud retrieval from INSAT series of satellite.

The commitment of snow at the focal point effect will be higher, the commitment will drop more far from the focal point of effect. Shear warpping shears the image in 2D and warp the intermediate image in the end. Texture Based slicing generates the immediate slicing while rendering.

III. BACKGROUND

Traditional volume rendering approaches were texture slicing in 2D and 3D. The downside of using texture slicing in two dimensions was: absence of the tri linear interpolation amongst 2D slices. Where in texture slicing with three dimensions, due to the perspective projection, differences were there between volume samples distance, per ray. It got fixed by using the spherical shells but that added the need for more complicated geometries. Above mentioned approaches mainly show visual artifacts. Compared to ray casting methods, it has less flexibility [4].

IV. RAY CASTING

The ray casting algorithm collects the details of opacity and color along the ray traversing path through the entire volume. Ray starts from the eye (viewpoint), it traverses mostly every voxel in the dataset at a particular inclination. To record the final results, a two dimensional image plane with the same size of one input slice placed within the volume data & viewpoint [5].

As shown in (1), this technique replicates an interaction between the volume (participating media) and the light. Here, values $\tau(x(\lambda))$ and $c(x(\lambda))$ represent the absorption and the color of a sample at a distance λ, parameterization of the ray has been represented by $x(\lambda)$

$$C = \int_0^d c(x(\lambda))\,\tau(x(\lambda))\, e^{\int_0 \tau(x(\lambda'))\,d\lambda'}\, d\lambda \qquad (1)$$

As shown in (2), the equation of volume rendering can be approximately discretized, where c_i and α_i respectively represent the color and the opacity of the i^{th} sample.

$$C \approx \sum_{i=0}^{n-1} \prod_{j=0}^{i-1} (1-\alpha_j)\alpha_i c_i \qquad (2)$$

This algorithm directly evaluates the equation of volume rendering and achieves good quality visual results. Moreover, the process can be parallelized easily because for each ray, the calculation is independent [4].

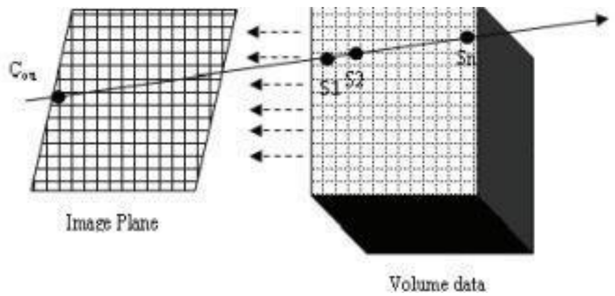

Fig. 1. Ray-casting

V. ANALYSIS OF VARIOUS APPROACHES TO RAY CASTING ALGORITHM

A. Ray Grouping approach for Visualization of Medical Data

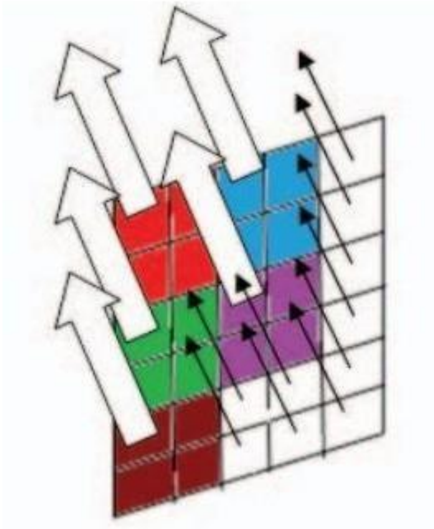

Fig. 2. Ray grouping

Here, the objective is to simplify the ray casting algorithm. First of all, rays which are cast from the similar voxels are grouped here. These groups are verified on every successive sampling based on measures of similarity. While data preparation, the ray gets sampled at different locations. To obtain the resulting values, those sample points are interpolated. So over the whole volume, the absolute interpolations took place will be (mns). To reduce this computational complexity, the rays that originated from the similar successive voxels are grouped. This procedure makes an isolated representative ray to correlate to all of the group members.

For grouping, intra block variance mechanism is used here as a measure to maintain the group of voxels. First the initial input slice is divided into smaller blocks of the same size. The variance within each of those blocks is then measured

The user defined threshold is compared to the computed variance value. If the computed variance lies in the threshold range, every voxel in the block gets grouped into a single unit. From that group, just one representative ray is casted instead of casting individual ray from each voxel. If a deviation is found in the variance, then the voxels will be ungrouped and the rays will get casted from individual voxels. This idea is implemented in Fig.2. Solid arrows from the shaded voxels represent rays for those voxels. Each shade shows different groups. When it comes to showing the object, the information of grouped and non-grouped voxels are both considered so information would not be lost but the better similarity measure can be found in future. Ray grouping assures the computational complexity reduction. It also preserves the quality of the image [6].

B. Mouse Picking approach for 3D Spatial Information Open-platform

The aim here is to put forward a method called mouse picking. It selects the desired geographical information for users when three dimensional geospatial information gets visualized around the globe. For this research, the dataset of a good quality 3D national spatial information is provided by VWorld Data Center operated by the Rep. of Korea's Ministry of Land. The dataset of spatial information data gets built on a pretty big scale. On a three dimensional map, to select a picking point precisely, a huge amount of calculations are needed. Earth surface is primarily segmented into fifty tiles provided that tiles started defining from level 0. Tiles of level 1 are composed of a total 200 tiles because every level 0 tile is further segmented into 4 tiles. Total 70 billion tiles are there to minimize. After minimization the algorithm starts working. First, the direction vector (d) is calculated. It connects the camera position and the selected a point by a user which is illustrated in Fig. 3. Then it calculates the closest three dimensional object of spatial information which collides with the direction vector (d) from the position of the camera.

First, they calculated if there are any intersections of the direction vector (d) is there or not with the tile cubes represented on the screen. A function is defined to calculate the intersection point between the direction vector (d) and all polygons of a 3D model. Then, it checks and calculates if there is any intersection of direction vector (d) with all the building cubes in the colliding tile cubes are there. 'collisionPoints array' is defined to store all the colliding intersections.

At last, the Collision Points array gets sorted in the ascending order. The most immediate intersection is picked by the user. For classification of tiles which will be presented on the screen, quad-tree based tile search method is used. WebGL (Web Graphics Library) is used for implementation of this approach. The proposed idea reduces the total number of targeted geospatial information models. It also reduces the calculating time [5].

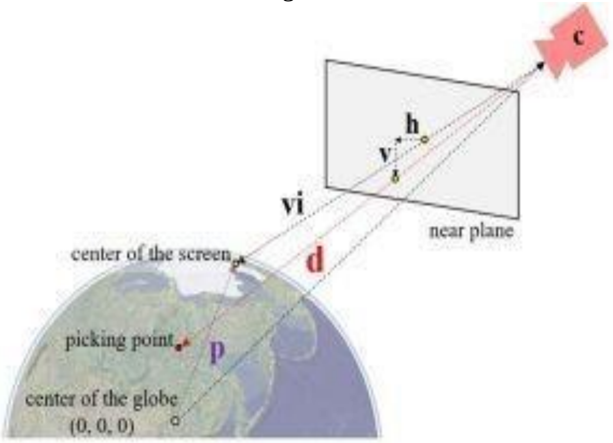

Fig. 3. Mouse Picking with ray casting

C. A Novel Parallel Ray-Casting Algorithm

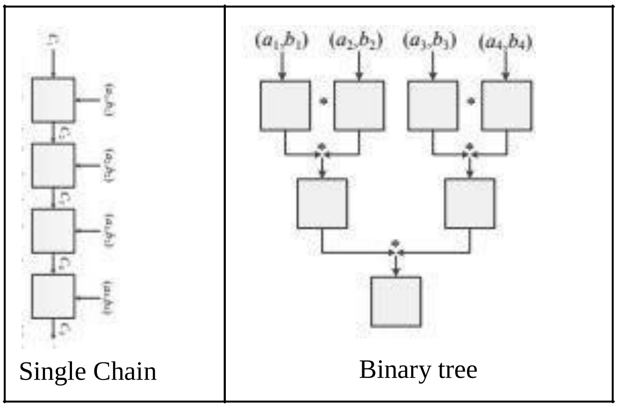

Figure 4: Traditional vs. Novel Parallel Ray-casting

Here, the novel approach to the ray-casting algorithm is defined based on the traditional ray casting system. In conventional ray casting algorithm, the opacity and color of (i-1)th voxel is used to determine the color and opacity of voxel i. A star operation is defined here as a novel operation, and it can be computed parallelly in the proposed algorithm. It is compared with the serial single chain of star operations in the ray-casting algorithm. In this new approach a star operation (∗) can be defined as,

$$(a_{[i,j]}, b_{[i,j]}) = (a_{[i,m]}, b_{[i,m]}) * (a_{[m-1,j]}, b_{[m-1,j]}) = (a_{[i,m]} \cdot a_{[m-1,j]}) + (b_{[i,m]})\ (i > m > j)$$

Here, associative law is defined and conventional ray-casting is now upgraded to novel parallel ray casting. In the conventional ray casting algorithm, only one-star operation unit used where in this novel approach, n/2 star operation units are implemented. It can achieve the reduced computation complexity of O(logn/log2). The computational complexity of traditional algorithm was O(n) and now it is reduced to O(logn/log2). Because of one star operation in conventional ray casting algorithm, chain-like structure appears where in novel parallel algorithm, binary tree like structure appears [7].

D. Volume Ray Casting using Different GPU based Parallel APIs

To implement a ray-casting algorithm, a bunch of parallel Application Programmable Interfaces (APIs) can be used. In order to help developers in selecting one of the APIs, it is necessary to analyze the outcome of ray casting algorithms in various APIs. Here, the performance comparison amongst Open Computing Language (OpenCL), OpenGL with fragment shader, CUDA and OpenGL with compute shader are presented. While traversing the object, ray intersects with entry and exit points. This type of intersection can be normally calculated using these two approaches: 1) rasterization 2) ray/box intersection test.

For obtaining the entry point (first hit of the ray), back face culling is used and for obtaining the exit point (last hit of the ray) front face culling is used in the method of rasterization based intersection. That is feasible to calculate the direction of the ray, origin of the ray and end of the ray using these 2 values.

The intersection (volume/ray) for every pixel of a framebuffer is generated here. In the ray/box intersection, basically the divide & conquer approach is used. Intersection of the ray with a box can be deducted to the intersection of a ray with every slab in the volume. Fragment shader is a part of Open Graphics Library(OpenGL) and works with 2 rendering passes. Direction and length of volume is determined while traversing the object. Here, with an early ray termination process, the code gets optimized; which means that whenever the ray assembles the opacity value higher than its threshold (which is also called opacityThreshold), the loop ends. After that, the resulting opacity and color get stored in the framebuffer which is presently attached. Just like the fragment shader, compute shader also permits the texture access using samplers. There is just 1 difference with texture of output which has to be connected with glBindImageTexture function as the image. OpenCL is a different API than OpenGL. For rendering, the OpenGL textures have to be binded to an OpenCL image. At each thread, function gets executed only once where every thread gets identified uniquely with its variable. Without further extra overhead, CUDA accesses the OpenGL output texture. Because of that the OpenGL texture can possess the same GPU memory location of the CUDA surface or texture.

The experiments were tested using datasets starting from 16MB to 600MB, each volume tested with different block sizes (4 to 128 threads per block) & transfer functions. The best overall results of our tests were obtained by combining compute shader and ray/box intersection test. Compared to rectangular blocks, squared ones present superior results, because squared configurations can make more coalesce accesses to any volume texture. Larger datasets with few threads is a better option and vice-versa [4].

E. 3D-visualization of carbonate rock pore facies applications with ray casting

Carbonate 3D geological reconstruction method is presented here. This method is built on the exclusive combination of ray casting nearest-neighbor interpolation algorithm, imaging logging data, and Graphics Device Interface double buffering technique. In practical application of gas and oil exploration, this technique can be applied. The experimental results illustrate that 3D images of carbonate rock pore faces need to be very clear. This method can help researchers observing the distribution of carbonate rock pores. It also enhances the physical properties of the rock more clearly and conveniently.

The aim represents a technique to reconstruct the three dimensional geological pores of carbonate rocks. In world coordinates, 3D image display position can be found by calculating the value of gray gradient of the voxel with Image logging technique. In the data preprocessing phase, traversal in the three-dimensional data field is done based on voxel. For light projection algorithm, the computational efficiency was enhanced. In the next phase, this system acquires the nearest neighbor interpolation algorithm according to the research object requirements. In order to calculate all the samples of opacity value and grey value, the light intensity is used. As per the properties of carbonate rocks, GDI bitmap is encapsulated by GDI graphics object classes and member function were provided to load and operate the bitmap. After that, encapsulated GDI palette saves the rules by using 3 colors obtainable information. In the carbonate rock, the skeleton phase, connected phase, and pore facies are represented by them respectively. Finally, using the double buffer technique, this system gives outcomes of visual 3D images. The user (observer) can see those 3D images with 360 ° rotation & translation. Looking from any direction {x, y, z}, the observer witnesses the distributed formation of the rock interface.

Calculation of dataset as well as the researchers study is very large, thus affects the speed of the 3D image display. GDI double buffer technique is used to enhance the speed of the displaying image. Results illustrate that GDI double buffer technique can constructively facilitate better observations and better research. The increase rate in display efficiency after optimization is 38.9%. The accuracy is enhanced this way [8].

F. Visualizing 3D atmospheric data with spherical volume texture on virtual globes

In this research, spatial data that covers a thin but broad geographic area is visualized using volume ray-casting. As a remedy of under-sampling, over-sampling and accuracy loss, virtual globe can be directly created. In this research the Geographic coordinate System (GCS) is converted to Cartesian coordinate system (CCS) because latitude, longitude and altitude are not easy to deal with.

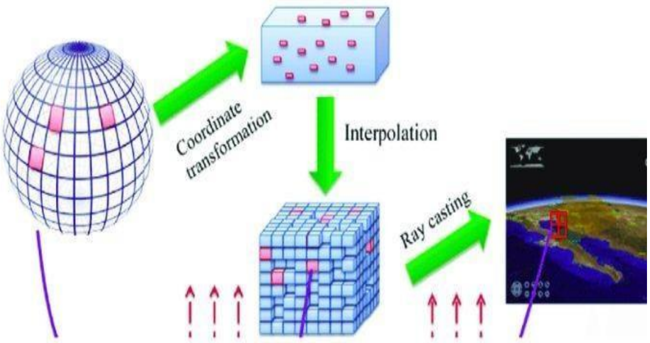

Fig. 5. Ray-casting on virtual globes

The volumetric ray-casting technique is executed in the different rendering framework. It integrates effects of volume into a virtual globe made up of a diversity of background geospatial objects of data, like terrain [15], vector shapes, imagery, and three dimensional geometric models. The dataset used here is The Hurricane Isabel dataset which is provided by National Center for Atmospheric.

Research (NCAR) and the U.S. They present a volume ray-casting to achieve the following: (1) On virtual globes, the direct visualization of GCS-based volume data is done without resorting to techniques of reprojection and resampling of data to provide more useful representation of the actual dataset contents as they were captured; (2) For more appropriate observation of elevational variations, Real-time vertical scaling took place in GCS (3) For providing improved geographic backgrounds for interactive volume analysis, seamless integration of atmospheric data with other type of data contents on the virtual globes is done. They summarized that (1) The volumetric ray-casting can easily provide on-the-fly altitudinal scaling. (2) The deferred rendering pipeline can combine different volumetric effects with a virtual globe. Implementations are conducted to assess the FPS (frames per second) for computation concentrated visualization systems. Performance of rendering can be declined easily with either decrement of viewing distance or increment of vertical scaling when the viewing distance is large. As the viewing point gets closer, the rendering performance gets more sensitive to the viewing distance than to the vertical scaling factor [9].

G. Highly Efficient and Speedy Algorithm for Volume Rendering

Ray-casting with 3D visualization is widely used in medical imaging , but the speed of rendering is quite slow. To enhance that speed, based on rapid interpolation & space leaping a highly efficient and fast ray casting is presented here. In Ray-casting, ray travels either in an empty or a non-empty voxel. An empty voxel generally makes zero contributions to the results but it still affects time and calculation. In this research, a space leaping method is acquired to skip empty voxels. Whatever the direction is, for each empty voxel, it records the closest distance to an opaque voxel, and considers this distance as a forward step length. Until it finds another non-empty voxel, this process will be repeated. So, the empty voxel goes through this jumping process and it contributes nothing to the rendering result. Now-a-days, accelerated algorithms use tri-linear interpolation for computing transparency & color, then they gather opacity and color of each sampling point to draw the 2D images with its 3D effect. Sampling needs to be done in interpolation. Every interpolation requires 19 additions and 24 multiplications for the process of resampling, which can get denoted as T= (19T1 + 24T2) * m * n, where m denotes voxel, and n is the number of times sampling is done. For an accelerated approach, let’s assume a ray intersects a single voxel at 2 different points M & N, then it gets sampling points at the line MN. Here, Vi is value of sample, i=1…k

$$Vi = Vm \cdot \frac{Vn - Vm}{k+1} \cdot i$$

This paper presented one process of resampling which just needs to perform the linear interpolation two times. The calculation time of resampling procedure with m voxel with n resampling can be denoted as: T= [(19+n) T1 + (24+2n) T2]*m, where T1 denotes the time for addition and T2 denotes the time for multiplication. Here, the computational complexity is reduced and not only it is producing a good quality rendered image, but it also improves rendering speed to a higher level [10].

H. Three-dimensional Reconstruction of Coarse grained Soil Fabric Based on Improved Ray Casting Volume Rendering Algorithm

Most two-dimensional fabric of coarse-grained soil models are not accurate, the three-dimensional model is needed. An algorithm based on the improved ray casting volume rendering is proposed for reconstructing the three-dimensional model of the fabric of coarse-grained soil. This research represents an enhanced algorithm by performing adaptive segmentation of threshold while preprocessing of original data field which is comprised of CT images, while applying ray casting for volume rendering. In this approach, most of the meaningless data gets filtered, and only the target data gets reserved, which is, the coarse-grained soil data. This process reduces the amount of data that goes for reconstruction, it also enhances the ray casting algorithm’s efficiency. Here, to determine, the optimum threshold, simulated annealing algorithm gets applied to find the maximum between-cluster variance. With this approach, transparency is shown and the reconstruction of soil fabric is done. In Matlab 6.5, the experiments are done. Results show that the applied algorithm enhances the visual effects and the efficiency of reconstruction, it fulfils the application requirements better. This approach reduces the time of reconstruction for volume data and effectively enhances the quality of rendering [13].

VI. CONCLUSION

Here, different approaches to ray casting are enlisted. All methods are applied to improve Ray-casting in order to achieve the desired outcomes. In medical imaging, volume rendering has been a key to achieve best 3D visualization of body parts. For different applications in different fields, ray-casting is applied for better visualization. We can see that as an image order technique with application wise improvements, it shows images with better quality and less complexity. In future, Volume rendering of weather data received by satellites and radars can be done on the web using ray-casting. As forecasting of weather events needs to be accurate, the volume rendering can help show better results. Showing intermediate results of 3D visualization, would specifically show weather behavior at each level of atmosphere, this can also be achieved in future for better forecasting results using volume rendering.

ACKNOWLEDGMENT

We would like to pay our gratitude to the Space Applications Center, SAC, ISRO. For providing resources as well as guidance. We are also thankful to Dr. Hiteishi Diwanji, HOD at LD College of Engineering for her constant support throughout this journey.

REFERENCES

1. Dubey, Rashmi, Sarika Jain, and R. S. Jadon. "Volume Rendering: A Compelling Approach to Enhance the Rendering Methodology." 2016 Second International Conference on Computational Intelligence & Communication Technology (CICT). IEEE, 2016.
2. Callahan, Steven P., et al. "Direct volume rendering: A 3D plotting technique for scientific data." Computing in Science & Engineering 10.1 (2008): 88-92.
3. Binyahib, Roba, et al. "A scalable hybrid scheme for ray-casting of unstructured volume data." IEEE transactions on visualization and computer graphics 25.7 (2018): 2349-2361.
4. Sans, Francisco, and Rhadamés Carmona. "Volume ray casting using different GPU based parallel APIs." 2016 XLII Latin American Computing Conference (CLEI). IEEE, 2016.
5. Lee, Ahyun, and Insung Jang. "Mouse Picking with Ray Casting for 3D Spatial Information Open-platform." 2018 International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2018.
6. Mehaboobathunnisa, R., AA Haseena Thasneem, and M. Mohamed Sathik. "Ray grouping based ray casting for visualization of medical data." 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16). IEEE, 2016.
7. Y. Zhang, P. Gao and X. Li, "A novel parallel ray-casting algorithm," 2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, 2016, pp. 59-61.
8. Zhang, Kui, Lingchen Zhu, and Lin OuYang. "Application of 3D-visualization of carbonate rock pore facies based on ray casting algorithm." 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA). IEEE, 2017.
9. Liang, Jianming, et al. "Visualizing 3D atmospheric data with spherical volume texture on virtual globes." Computers & Geosciences 68 (2014): 81-91.
10. Shicai, Yu, Wu Qianjun, and Lu Rong. "An high efficient and speed algorithm of Ray Casting in volume rendering." 2011 International Conference on Consumer Electronics, Communications and Networks (CECNet). IEEE, 2011.
11. Luo, Jianxin, et al. "An efficient Ray Casting method for terrain visualization." 2011 International Conference on Multimedia Technology. IEEE, 2011.
12. Tan, Jianhao, et al. "Design of 3D visualization system based on vtk utilizing marching cubes and ray casting algorithm." 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). Vol. 2. IEEE, 2016.
13. Wenli, Yang, Zeng Zhiyuan, and Yang Li. "Three-Dimensional Reconstruction of Coarse-Grained Soil Fabric Based on Improved Ray. Casting Volume Rendering Algorithm." 2009 International Forum on Information Technology and Applications. Vol. 2. IEEE, 2009
14. Kim, Taeho, and Jinah Park. "Analyzing 3D cell data of optical diffraction tomography through volume rendering." 2018 International Workshop on Advanced Image Technology (IWAIT). IEEE, 2018.
15. Luo, Jianxin, et al. "An efficient Ray Casting method for terrain visualization." 2011 International Conference on Multimedia Technology. IEEE, 2011.

AUTHORS PROFILE

Ms. Palak Chatwani, pursuing M.E. at L.D. College of Engineering and doing her dissertation at DWD/MRG/EPSA SAC, ISRO.

Mrs. Shivani Shah, Scientist-SF at DWD/MRG/EPSA at SAC, ISRO. She works as a MOSDAC administrator as well as principal investigator cloud retrieval from INSAT series of satellite. She has 10 research publications. She is in charge of software web based applications for weather and ocean products. She has won three team excellence awards, an award for Megha Tropics and an award for iNSAT Meteorological Data Processing System IMDPS.

Dr. Purvi N. Ramanuj, Gold Medalist in ME (CST), PhD, is working in the department of Information and technology, L. D. College of engineering. Her research area is security. She has published more than 25 research papers. More than 19 years of experience in industry as well as teaching. Currently four students are pursuing PhD under her guidance.