CSE 566 Virtual Reality, Spring 2020, Individual Final Project: Accelerated Stereo Rendering with Hybrid Reprojection-Based Rasterization and Adaptive Ray-Tracing

Presenter: Tianao Wang, SBU ID: 112819772,

Paper author(s): Niko Wißmann(TH Koln, Computer Graphics Group), Martin Misiak(TH Koln, Computer Graphics Group, University Wurzburg, HCI Group), Arnulph Fuhrmann(TH Koln, Computer Graphics Group),Marc Erich Latoschik(University Wurzburg, HCI Group),

Paper session: Session 26: Visual rendering

1 Session Summary

In the session, many authors share their methods to do visual rendering. This paper's author briefly introduces what the whole paper and project do. VR rendering is requested to be stereoscopic, larger FOV, higher resolution and higher FPS and normal desktop rendering. So, VR rendering need more time to do the work. The authors' approach is aimed to accelerate the stereoscopic rendering. There are three creative steps of their work. First is do reprojection using adaptive grid warping. Second is implement a perceptually motivated subdivision heuristic. The last is correct reconstruction of reprojection errors using ray-tracing.

The author shows an overview of the rendering system which I will detailly talk about in the summary of paper. Here I just briefly introduce what the author said in the session about the system. First, the author says that they use grid mesh to do the reprojection. The more mesh we use, the more detailed view we can get. They get the view of the left eye and gain the colour and depth. Then, they use a n*n grid mesh to representation the view and displace the grid mesh to record the information. After this, they can use reprotection to render grid from right eye. Second, they need to know which mesh is needed to be subdivided. In order to solve this problem, they implement a heuristic way to divide using binocular parallax. Third, after reprojection, they have to detect the disocclusion and discard these regions. Finally, they use hole filling to get the correct pixels of these regions. They choose to shaders to reconstruct the missing fragments. After all these steps, they get the view of right eye. The author also does some test to show the advantage of their algorithm.

2 Summary of the Paper

Stereoscopic rendering is a prominent feature of virtual reality applications to generate depth cues and to provide depth perception in the virtual world [1]. In the easiest way, we can render two views of two eyes individually. However, obviously, this way is not efficient. This paper is aimed to provide a hybrid rendering system to accelerate the stereoscopic rendering process of VR.

The figure1 shows the structure of system. This system is called hybrid rendering system because it uses standard deferred rendering pipeline to render the image of left eye. Then it uses a 3D grid warping approach to generate a reprojection image of right eye. After this, the system detects the reprojection errors and use an adaptive real-time ray-tracing hole-filling technique to fix these errors. Here I want to talk about some important technologies in this system.

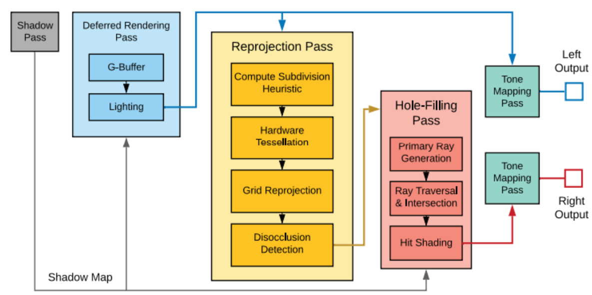

Figure 1: Overview of the hybrid rendering system, consisting of a standard deferred rasterization pass for the left viewpoint and a subsequent reprojection pass with ray-traced hole-filling for the right viewpoint.

2.1 3D Grid Warping and Subdivision Heuristic

The system uses 3D grid warping to record information. The warping grid is initialized at the system's start in NDC coordinates with a cell size of 16×16 pixels and bound to the GPU [1]. The author wants to use the left eye view to do reprojection and change it into right eye view.

However, 16×16 pixels are not enough to represent all the geometry surfaces. So, we need to figure out which grid is need to be subdivided. In this paper, the author implements a perceptually motivated heuristic based on binocular parallax. Binocular disparity refers to the difference in image location of an object seen by the left and right eyes, resulting from the eyes' horizontal separation (parallax). The brain uses binocular disparity to extract depth information from the 2d retinal images in stereopsis [2]. Binocular parallax is the angle θ between the lines of sight of both eyes, separated by an interpupillary distance ipd, when converged on a point at distance z [1].

$$\theta(z) = 2\tan^{-1}\left(\frac{ipd}{2z}\right) \quad (1)$$

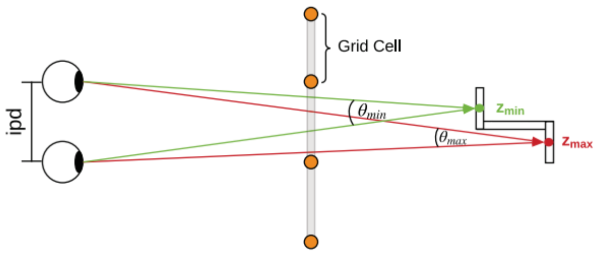

Figure 2: Grid subdivision heuristic using binocular parallax. We can get the θmin and the θmax which represent the closest point and farthest point the two eyes can see. If two angles differences are above the threshold, we can consider these cells are needed to be divided.

$$\theta_{\Delta}(z_{min}, z_{max}) = \theta(z_{min}) - \theta(z_{max}) \tag{2}$$

We use the $\theta_{\triangle}$ (z_{min}, z_{max}) to represent the threshold. For the values under this threshold, our eyes cannot distinguish the difference of depth. So, there is no need to do division. Only do the division for the values which is above this threshold can save quite amount of space and time for GPU to do the rendering.

2.2 Hardware tessellation

In computer graphics, tessellation is used to manage datasets of polygons (sometimes called vertex sets) presenting objects in a scene and divide them into suitable structures for rendering [3]. After subdividing by Heuristic method, the system uses tessellation to adapt the grid cell size to the underlying source image. The tessellation is done via the hardware tessellator stage introduced in DirectX 11, which can be controlled by specific hull and domain shaders [1]. With the subdivided grid warping, the author uses a reprojection matrix to get the image of the right eye.

2.3 Disocclusion Dection

The reprojection process has some errors because there are some areas where can be seen by the right eye but cannot be seen by the left eye. So, these areas are flagged and will be corrected in the next step.

To this end, a Sobel operator is used in the domain shader, to filter the input depth buffer at a given vertex position. The resulting pervertex depth gradient is passed on to the fragment shader, where a threshold condition decides if a fragment should be discarded.

2.4 Hole-Filling

Now author has these areas where have errors and the author uses a hole-filling process to correct the errors.

The author chooses to use ray tracing to do the hole filling job. There are three shaders in the hole-filling process. A ray generation shader as the initial entry point to start the ray-tracing. A closest hit shader used for the actual hit point shading and a miss shader, in case no geometry is hit [1].

Unlike rasterization, where pixels are shaded within a 2 × 2 pixel block and four differentials can be generated to compute an appropriate texture MIP-level, this cannot be done by a single ray. One solution is to use cone tracing [4]. Using the projected footprint of the cone, the texture MIP-level can also be calculated for a single hit point of a ray [1].

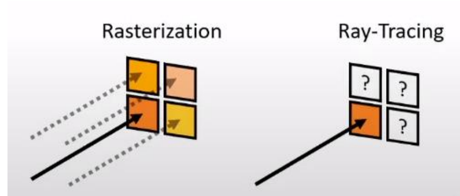

Figure3: ray tracing cannot shader the pixels around the ray.

2.5 PERFORMANCE ANALYSIS

Three rendering systems have been tested in the performance analysis for four different scenes. The naive stereo rendering implementation, using two full render passes, serves as a reference system and is referred to as Plain Stereo. The presented reprojection system with hybrid rasterization and adaptive real-time ray-tracing is called Repro RT, while the same system but with rasterization-based hole-filling is called Repro ReRaster[1] .

Figure4: four testing scenes.

Table 1: Test scene properties.

Scene	Triangles	Textures	Texture size	Lights
Bistro In	1,022,521	211	1 × 1 to 2048 × 2048	1 directional and 13 point
Bistro Ex.	2,832,120	405	1 × 1 to 2048 × 2048	1 directional
Bistro Night	3,845,871	607	1× 1 to 2048 × 2048	1 directional and 31 point
Temple	606,376	148	512 × 512 to 2048 × 2048	1 directional and 13 point

Table 2: Total frame time in ms for each rendering system, measured over 1000 frames. The speedup factors are in comparison to the Plain Stereo renderer.

Resol ution	Scene	Renderer	Total time [ms]	Speedup factor
HD	Bistro In	Plain Stereo	3,074	
		Repro ReRaster	2,778	1.11
		Repro RT	2,404	1.28
	Bistro Ex.	Plain Stereo	5,485	
		Repro ReRaster	5,878	0.93
		Repro RT	3,779	1.45
	Bistro Night	Plain Stereo	7,602	
		Repro ReRaster	7,989	0.95
		Repro RT	5,250	1.45
	Temple	Plain Stereo	3,747	
		Repro ReRaster	2,912	1.29
		Repro RT	2,545	1.47
4K	Bistro In	Plain Stereo	11,673	
		Repro ReRaster	8,700	1.34
		Repro RT	8,297	1.41
	Bistro Ex.	Plain Stereo	12,901	
		Repro ReRaster	9,719	1.33
		Repro RT	8,638	1.49
	Bistro Night	Plain Stereo	18,989	
		Repro ReRaster	13,771	1.38
		Repro RT	12,215	1.55
	Temple	Plain Stereo	14,598	
		Repro ReRaster	10,392	1.40
		Repro RT	9,409	1.55

2.6 Conclusion

According to the test, the hybrid system performs the best in the three systems. Showing that the system achieves a speed up for stereo rendering.

3 CRITIQUE OF THIS PAPER

3.1 pros and cons

Pros:

1. This render system uses the image of left eye to do reprojection into image of right and uses ray tracing to fix the errors. This can highly save time and space than normally render two images of both eyes.

2. This paper's technology doesn't depend on any specific rendering pipeline. So, it can be used for different GPUs with different rendering pipelines.

3. The render system performs better in high resolution scenes which means it may have a large application prospects in the future.

Cons:

1. This render system mainly test on static objects, doesn't show the performance on dynamic objects.

2. User's eyesight will be limited due to the heuristic subdivision because some information is discarded.

3. the performance of rendering depends mostly on the number of ray-traced pixels. Therefore, in a scenario with massive disocclusions, ray-tracing could lose its advantage over rasterization.

4. the reprojection does not produce correct specular reflections as we reproject pixel values under the assumption of a Lambertian surface

3.2 The relationship between the paper and the topics covered by CSE 566

There are many parts in the paper which are related to the topics we covered in CSE566.

First, in the class, we talked about the refresh rate of the VR. It is mainly from 60Hz to 90Hz. So, if a scene has too much triangles and textures, it will take a large amount of time to do the render, specially for VR because we need to render for both eyes. Using this system can accelerate the render process which can improve the reality of VR.

Second, in slide 3.2, we learned that we can add two matrixes to generate the images for left and right eye. As far as I am concerned, this is close to the idea of reprojection in the paper. In the paper, It use the image of left eye to generate the image of right eye using a reprojection matrix.

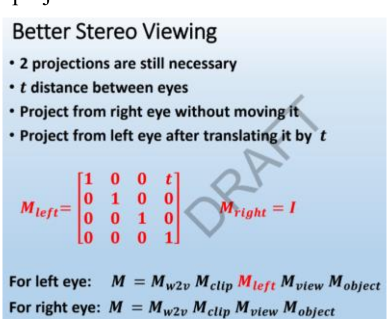

Figure5: slide 3.2 screenshot

Third, in the slide 7.1, we learned that we can use two GPU to render two eyes' images in order to satisfy the high requirement of high refresh rate. In this paper, it provides another solution that we can use the reprojection method to accelerate the render when we don't have enough equipment.

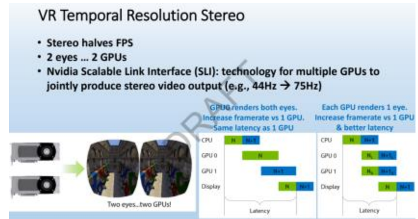

Figure6: slide 7.1 screenshot

Last, in the slide 7.2, we have learnt that there is stereo depth cue for our eyes which is corresponding to the grid subdivision heuristic using binocular parallax in the paper. With this phenomenon, we can let some pixel in the same depth when they are in the range of angle threshold. Also, this explains why we will get some areas which cannot been rendered using reprojection.

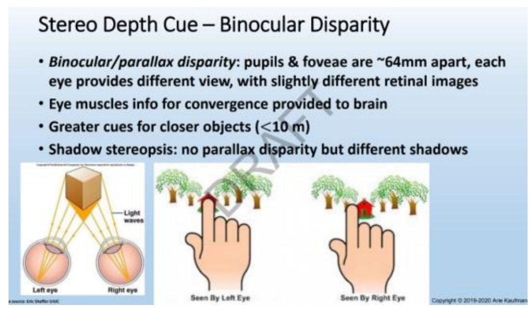

Figure7: slide 7.2 screenshot

3.3 Future work

The possible future work you would like to suggest based on your understanding on this paper and on the relevant domain.

First, I want to find is there other better way to do the perceptually motivated grid subdivision heuristic which is very important in this paper.

Second, I want to test and try to render for the dynamic object in the scenes. The paper mainly considers on the static object in the scenes. I would like to see whether there is some difference between dynamic objects and static objects.

Last, I want to find a better way to do an efficient handling of semi-transparent geometry and perceptually correct specular reflections which is cannot be done by this system.

REFERENCES

[1] Niko Wißmann, Martin Misiak, Arnulph Fuhrmann, Marc Erich Latoschik. Accelerated Stereo Rendering with Hybrid Reprojection-Based Rasterization and Adaptive Ray-Tracing. Proceedings of 27th IEEE Virtual Reality Conference (VR '20), Atlanta, USA, to be published

[2] Wikipedia definition of binocular disparity. https://en.wikipedia.org/wiki/Binocular_disparity,2020.

[3] Wikipedia definition of Tessellation. https://en.wikipedia.org/wiki/Tessellation_(computer_graphics) ,2020.

[4] J. Amanatides. Ray tracing with cones. SIGGRAPH Computer Graphics, 18(3):129–135, 1984. doi: 10.1145/964965.808589

4 Screenshot of session