A Review of Efficient Interactive Rendering of Natural Scenes Suitable for VR

Soo Kyung Ahn
ahnsoo002@myuct.ac.za
University of Cape Town
South Africa

ABSTRACT

Being in natural environments is important to reduce stress, and a simulated virtual nature environment is a viable alternative for this purpose. When rendering virtual environments, there can be problems that make the experience less enjoyable such as low frame rates. Low frame rates breaks immersion and thus is undesirable. Efficient rendering is of importance to reduce this problem. However, we would still like to have the images rendered to be as realistic as possible, so interactive rendering, or real-time updates of the rendered image based on changing scenes, is desirable. There are various techniques for rendering to do this, among which we discuss billboarding, level of detail, point-based rendering, volumetric textures, and skyboxes.

CCS CONCEPTS

• **Human-centered computing → Virtual reality**.

KEYWORDS

virtual reality, natural scenes, real-time rendering

1 INTRODUCTION

Humans have made many technological advancements through the ages, and urbanization was one of the events that took place with it. As more people moved away from rural areas into urban areas, it meant that fewer people were having regular contact with nature. Contact with nature is important as it has an abundance of characteristics that are effective for restoration [14], improving mood and cognitive capabilities [4], and improving health and reducing general mortality [21].

Thus, the importance of being present in natural environments is well established, but some people may not be able to readily access real nature due to poor proximity [1], the safety of such areas [19] or physical disabilities [6]. In these cases using virtual nature is a viable alternative to real nature, providing some of the benefits of real nature [23].

The creation of such a virtual environment requires modelling it. To see the virtual environment, one must render it, or generate an image from the model. Another important aspect of the experience would be whether a user can interact with the virtual environment, or receive some kind of feedback from performing an action.

There are various ways to do this, and this paper discusses some of the ways in which this has been accomplished, focusing specifically on the efficient and interactive rendering of these virtual environments. Efficient rendering is important because it helps provide a more immersive experience from the increased frame rates [18]. This is especially of importance to rendering virtual environments as there is an additional overhead rendering two images for the left and right components of head-mounted devices typically used for virtual reality viewing. Interactive rendering, or the real-time update of the rendered image based on changing scenes, is necessary for realistic rendering as well as efficiency.

2 RENDERING TECHNIQUES

We discuss some of the methods used for rendering forests below.

2.1 Billboards

Billboards are one of the most common techniques used for rendering forests, due to their low cost. They can either be a single image that represents a whole model, or a set of arbitrarily oriented images called *billboard clouds*, where 3D models are simplified onto a set of planes with texture and transparency maps [9].

Regardless of which type of billboards are used, they usually cause parallax problems. Some unwanted artefacts may occur depending on the implementation, such as a phenomena like *popping* and *ghosting*. Popping occurs when an always camera facing billboard implementation is used, and refer to one image coming in front of another as you rotate the camera around the axes of two close billboards. As these do not allow for dense forests this implementation is not often used in practice, instead the billboard implementation that typically gets used are multiple fixed textured quadrilaterals paired with scaled fading of the textured quadrilaterals that do not face the camera based on their angle to it. This removes the popping, but introduces the ghosting effect which refers to the duplication of features [8].

Billboard clouds, or a cluster of billboards, can be paired with some clustering algorithm to the vertices of trees to represent them using billboards. In such cases, the number of billboards involved is typically very large, and rendering massive scenes is not efficient as lots of textured quadrilaterals need to be rendered. For efficient rendering, the use of billboards is suitable where there are very few models for viewing in the camera. For example, it would be useful when the models that exist in the environment are few as in the case of sparse forests, or only a few trees are in the proximity of the camera [3]. Recent research by Argudo et al. uses this approach, mapping texture onto models for distant trees, but switching to billboard clouds for closer trees [2].

2.2 Level of Detail

Level of detail, or the complexity of a 3d model representation, needs to be adapted based on some criteria for the efficient rendering of images, and this technique is utilized in a majority of, if not all, recent virtual reality applications to make the rendering interactive and efficient. Rendering virtual nature involves rendering a combination of trees and terrain and we look at the two classes of

Figure 1: Example of the use of billboards clouds to represent different types of plants. Leftmost plants of each type are the original model, the center plant is an approximation of the model using a clustering model called k-means algorithm, the rightmost is the approximation using additional model information. [3]

LOD (level of detail) techniques. Figure 2 and 3 demonstrate the basic idea of LOD.

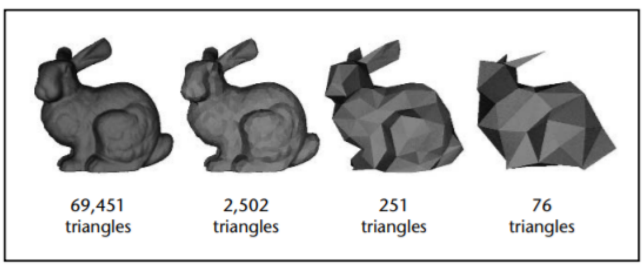

Figure 2: A model represented with different levels of detail. [17]

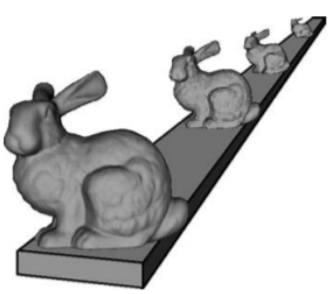

Figure 3: The level of detail of the model changing based on distance. [17]

2.2.1 Discrete LOD. Most applications today use the discrete LOD technique to render trees, creating multiple versions of every object with a different LOD during an offline preprocess. Then, the appropriate version of the object is selected at runtime based on some criteria such as distance, size, or priority. For example, the further the object, the coarser version of the object we can use, reducing the number of polygons and decreasing the rendering time. A limitation of the discrete LOD is that it cannot predict the viewpoint of the camera from which an object will be viewed from and hence the LOD is reduced uniformly. The discrete LOD technique is the preferred method for most applications because of the decoupling of the simplification process and the runtime rendering, taking away the simplification computation during runtime [17].

2.2.2 View-dependent LOD. On the other hand for terrain rendering, the view-dependent LOD is used. View-dependent LOD is an extension of continuous LOD which creates a data structure encoding a continuous spectrum of detail where the desired level of detail is chosen at runtime. The advantage of view-dependent rendering is that it selects the most appropriate level of detail according to the current view, allocating more polygons closer to the viewpoint and less polygons further away from the viewpoint. This is important to maintain high-fidelity images while preserving high frame rates especially for complex large models like terrain [17]. This reduction of the number of polygons rendered allows the efficient and interactive rendering of models.

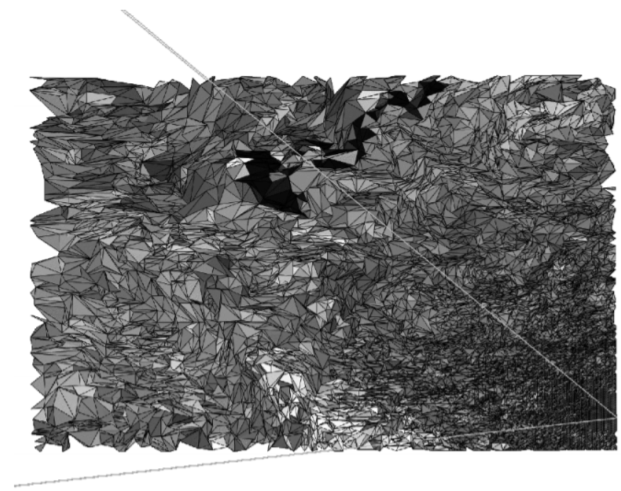

Figure 4: Birds-eye view of a terrain model demonstrating the simplification provided by view-dependent LOD technique. Two lines represent the field of view, and the model is displayed at full resolution near the viewpoint and is simplified the further away it is from the viewpoint. [17]

There are efficient algorithms that implement the view-dependent LOD technique to handle terrain rendering such as ROAM (Real-time Optimally Adapting Meshes), an algorithm for constructing triangle meshes optimizing view-dependent error metrics. It is very efficient because the execution time is proportionate to the number of triangles that change at each frame, and this number is typically small. [11]. More recently, CABTT (Cached Aggregated Binary Triangle Trees) extended the ROAM algorithm providing additional optimizations [15]. Some other approaches take divide the terrain into quadrilaterals [16] but there has not been extensive comparison of performance on this approach to algorithms like ROAM or CABTT that utilize triangular meshes to our knowledge.

2.3 Point-based Rendering

This technique is the use of primitive points and lines to represent models.

Its strength is that it is very convenient to use, as points can be added or removed, and merged easily to get a suitable level of detail. It is an efficient technique for rendering complex geometry and it has been used to render virtual nature [7, 10].

However, it is not a good use case for forests because forests are a type of sparse geometry and this technique works best for opaque objects [7]. The fact that forests are transparent (objects can be seen through branches) leads to the result that a lot of data projects to the same pixel. This combined with the fact that it is difficult to cull occluded data in advance is problematic for the performance. Another performance issue stems from the dependence on a large number of vertices, but the vertex transform rate is much slower compared to the fill rate [8] and using textures would be a rational alternative.

Research by Gilet et al. [12] on point-based rendering of trees shows that it is not free from artefacts such as popping and aliasing, and while the authors make a case of interactive efficient rendering enabling free movement of the viewpoint to view a single tree to an entire forest, they ignore the transparency of forests which does not address the problems mentioned previously and conclude by suggesting mapping textures could be more efficient for adding more detail while using less primitives.

2.4 Volumetric textures

The volumetric textures technique maps a 3D layer on a surface using a 3D dataset as a texture pattern, and is suitable for a layer of continuous vegetation covering a landscape. This is supported by hardware acceleration when the volume is rendered as textured slices, and has the advantage that parallax is perfect due to the proper depth, and texture hardware manages the filtering well. Fewer polygons are required to model the forest because each instance of the slice can represent multiple trees. However, care must be taken when slicing the volume so one cannot see between the slices at certain angles [8]. A limitation of this technique is that they cannot give precise control over specific elements of a model, and that it is meant for continuous terrain.

2.5 Skyboxes and Skydomes

The skybox is a cube in which the world is enclosed. A skydome is just a sphere or a hemisphere of a skybox. They are for backgrounds and has the property of being unreachable. They are very efficient because they often are fixed images that do not involve level of detail [24], but they can be modified for realistic rendering [20].

To reiterate, it would be suitable for the rendering the background of nature as done by Wang et al. [22] using skyboxes to generate scenes for nature while natural phenomena such as rain and snow were rendered in front of these scenes. While this could be enough for a very simple non-interactive virtual environment, if this is the only model that exists in the environment, the environment would be very limited as everything is unreachable. However, when used in combination with other rendering techniques to build a virtual nature environment, that could increase rendering efficiency while enabling exploration. In the case of rendering forests in combination with skyboxes, an suitable use would be to render mountains and celestial objects such as the sun, moon, and stars through the skybox while rendering forests using billboards.

3 DISCUSSION

See Table 1 for a side-by-side comparison of Papers.

We have looked at some of the techniques that are used in the efficient and interactive rendering of natural scenes, and we note that all techniques use the idea of level of detail to make simplifications along with some criteria, usually distance. Billboards would be useful when the viewpoint is fairly close to individual trees as the implementation is relatively simple and trees can be distributed arbitrarily. However, it is an expensive technique to be using to model an entire dense forest, and has some artefacts such as popping and ghosting. The use of point-based rendering methods are not feasible to render a large number of trees, but it would be useful for close views. On the other hand, the use of volumetric textures to represent forest patches reduces the number of polygons, but it is limited in the ability to control individual trees, control is limited to individual patches instead. Skyboxes could be well-suited for rendering backgrounds to display changing time for instance through the movement of the celestial bodies.

It seems that much of the research was considering the view of forests from an aerial point of view, which makes sense considering that they would be able to test the efficiency when a significant number of trees are in the view frustum. We hypothesize that it would be more efficient to view this from a first person perspective on the forest floor close to the trees, but research could be done to formally confirm this belief. Should this be the case, as well as the increase of computational power it would allow us to render more detailed, richer, and realistic forests as described by Guérin et al. [13], allowing one to automatically generate details like leaves, stones, and mushrooms, or even a mix of these three. Much of the research could be paired with a realistic lighting model as discussed by Bruneton et al. [5] for a more realistic virtual nature environment.

Figure 5: Generation of details that would be seen in a natural environment using leaves, stones, and twigs. [5]

4 CONCLUSION

When rendering a virtual natural environment, there are different techniques that would be suitable for different distances. In general, texture mapping techniques would be most efficient, albeit being

Table 1: Classification of Papers Concerning Natural Environment Rendering

Paper Title	Technique used	Suitable Viewing Distance	Lighting
Single-picture reconstruction and rendering of trees for plausible vegetation synthesis	Relief mapping and Billboards	Both far and close, respectively	Assume constant, fixed lighting, ignores leaves
Realistic Real-Time Rendering of Landscapes Using Billboard Clouds	Billboard clouds	Close	Locally accurate lighting, does not take into account other tree shades
Real-time realistic rendering and lighting of forests	z-fields, and shader maps	Far	accounts for view-dependent reflectance, slope-dependent reflectance, opposition effect, silverlining, sky illumination
Rendering forest scenes in real-time	Volumetric textures and aperiodic tiling	All distances (albeit limited for short distances)	Does not deal with dynamic lighting and shadowing
Interactive visualization of complex plant ecosystems	point and lines	Close	basic shadow map
Point-based rendering of trees	Point-based	Close	Estimates self shadowing
Real-time rendering of daylight sky scene for virtual environment	skybox	Very far (unreachable distance)	N/A

less controllable suitable for viewing from far. However, it could be paired with techniques like point-based methods or billboards for closer viewing. In the context of our project, we would focus on making the virtual nature more detailed and realistic, going beyond rendering trees and use automated methods for generating details, and giving a special concern to lighting as well.

REFERENCES

[1] Abdullah Akpinar. 2016. How is quality of urban green spaces associated with physical activity and health? *Urban forestry & urban greening* 16 (2016), 76–83.

[2] Oscar Argudo, Antonio Chica, and Carlos Andujar. 2016. Single-picture reconstruction and rendering of trees for plausible vegetation synthesis. *Computers & Graphics* 57 (2016), 55–67. https://doi.org/10.1016/j.cag.2016.03.005

[3] Stephan Behrendt, Carsten Colditz, Oliver Franzke, Johannes Kopf, and Oliver Deussen. 2005. Realistic Real-Time Rendering of Landscapes Using Billboard Clouds. *Eurographics* 24, 3 (2005). https://doi.org/10.1111/j.1467-8659.2005.00876.x

[4] Gregory N. Bratman, Gretchen C. Daily, Benjamin J. Levy, and James J. Gross. 2015. The benefits of nature experience: Improved affect and cognition. *Landscape and Urban Planning* 138 (2015), 41–50. https://doi.org/10.1016/j.landurbplan.2015.02.005

[5] Eric Bruneton and Fabrice Neyret. 2012. Real-time realistic rendering and lighting of forests. In *Computer graphics forum*, Vol. 31. Wiley Online Library, 373–382.

[6] Jim Burt, Sarah Preston, and Tom Costley. 2012. *Monitor of Engagement with the Natural Environment Survey (2009-2012).: Difference in Access to the Natural Environment Between Social Groups Within the Adult English Population.* Natural England.

[7] Carsten Dachsbacher, Christian Vogelgsang, and Marc Stamminger. 2003. Sequential Point Trees. In *ACM SIGGRAPH 2003 Papers* (San Diego, California) *(SIGGRAPH '03)*. Association for Computing Machinery, New York, NY, USA, 657–662. https://doi.org/10.1145/1201775.882321

[8] Philippe Decaudin and Fabrice Neyret. 2004. Rendering forest scenes in real-time. In *EGSR04: 15th Eurographics Symposium on Rendering*. Eurographics Association, 93–102.

[9] Xavier Décoret, Frédo Durand, François X Sillion, and Julie Dorsey. 2003. Billboard clouds for extreme model simplification. In *ACM SIGGRAPH 2003 Papers*. 689–696.

[10] O. Deussen, C. Colditz, M. Stamminger, and G. Drettakis. 2002. Interactive visualization of complex plant ecosystems. In *IEEE Visualization, 2002. VIS 2002.* 219–226. https://doi.org/10.1109/VISUAL.2002.1183778

[11] Mark Duchaineau, Murray Wolinsky, David E Sigeti, Mark C Miller, Charles Aldrich, and Mark B Mineev-Weinstein. 1997. ROAMing terrain: Real-time optimally adapting meshes. In *Proceedings. Visualization'97 (Cat. No. 97CB36155)*. IEEE, 81–88.

[12] Guillaume Gilet, Alexandre Meyer, and Fabrice Neyret. 2005. Point-based rendering of trees. In *Eurographics Workshop on Natural Phenomena*. Eurographics Association, 67–72.

[13] Eric Guérin, Eric Galin, François Grosbellet, Adrien Peytavie, and Jean-David Génevaux. 2016. Efficient modeling of entangled details for natural scenes. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 257–267.

[14] Stephen Kaplan. 1995. The restorative benefits of nature: Toward an integrative framework. *Journal of Environmental Psychology* 15, 3 (1995), 169–182. https://doi.org/10.1016/0272-4944(95)90001-2 Green Psychology.

[15] J. Levenberg. 2002. Fast view-dependent level-of-detail rendering using cached geometry. In *IEEE Visualization, 2002. VIS 2002.* 259–265. https://doi.org/10.1109/VISUAL.2002.1183783

[16] Yotam Livny, Zvi Kogan, and Jihad El-Sana. 2009. Seamless patches for GPU-based terrain rendering. *The Visual Computer* 25, 3 (2009), 197–208.

[17] David Luebke, Martin Reddy, Jonathan D Cohen, Amitabh Varshney, Benjamin Watson, and Robert Huebner. 2003. *Level of detail for 3D graphics.* Morgan Kaufmann.

[18] Alex Mackin, Fan Zhang, and David R. Bull. 2019. A Study of High Frame Rate Video Formats. *IEEE Transactions on Multimedia* 21, 6 (2019), 1499–1512. https://doi.org/10.1109/TMM.2018.2880603

[19] Gavin R McCormack, Melanie Rock, Ann M Toohey, and Danica Hignell. 2010. Characteristics of urban parks associated with park use and physical activity: A review of qualitative research. *Health & place* 16, 4 (2010), 712–726.

[20] Jason Mitchell, Moby Francke, and Dhabih Eng. 2007. Illustrative rendering in team fortress 2. In *Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*. 71–76.

[21] World Health Organization et al. 2016. Urban green spaces and health: a review of evidence. *World Health Organization: Copenhagen, Denmark* (2016).

[22] Changbo Wang. 2007. Real-time rendering of daylight sky scene for virtual environment. In *International Conference on Entertainment Computing*. Springer, 294–303.

[23] Matthew P. White, Nicola L. Yeo, Peeter Vassiljev, Rikard Lundstedt, Mattias Wallergård, Maria Albin, and Mare Lõhmus. 2018. A prescription for "nature" - the potential of using virtual nature in therapeutics. *Neuropsychiatric disease and treatment* 14 (08 Nov 2018), 3001–3013. https://doi.org/10.2147/NDT.S179038 30510422[pmid].

[24] Kuang Yang, Jiang Jie, and Shen Haihui. 2011. Study on the virtual natural landscape walkthrough by using Unity 3D. In *2011 IEEE International Symposium on VR Innovation*. IEEE, 235–238.