A Web3D Forest Geo-Visualization and User Interface Evaluation

Abstract

In this study, we developed a workflow for the construction of WebGL rendering and interactive visualization of forest landscapes from GIS and forest simulation datasets. We constructed a realistic virtual forest environment for our study site in Virginia. This Web3D virtual environment, built with X3DOM and HTML5, is capable of fast real-time rendering and walk-through simulation based on two Level-of-Details (LOD) switching techniques, and interactive over mainstream web browsers without a plugin. We explored the performance of a navigation aid and two types of LOD cut-off functions in the forest geo-visualization. We used a hunting game search task to test the effect of our user interface features. According to the paired t-test results, the navigation aid significantly increasing participants' performance, while the exponential cut-off and linear cut-off functions showed no significant difference from participants' perspective.

CSS Concepts: • **Human-centered computing → Geographic visualization** • Human-centered computing → Virtual reality; *Web-based interaction*

Keywords: Web3D, X3DOM, Level of Details, Forest Geo-Visualization, Virtual Environments

1 Introduction

Covering about 31% of global land surface (~4 billion hectares), forests play a significant role in the delivery of ecosystem services including biodiversity, climate regulation, carbon storage, prevention of soil erosion and flood mitigation[1]. Understanding forest structure, spatial patterns, dynamics, as well as growth of individual tree across the landscape is very important for sustainable forest management, public education, and decision-making processes [23]. The visualization of forest environment has been increasingly used to deliver the information about landscape patterns, forest stand conditions, and management plans [8, 9, 13, 19, 26]. Two-dimensional (2D) representations of forest landscape like satellite imagery or raster classified maps along with summary tables, or a list of individual forest stand components can depict forest characteristics. However, a three-dimensional (3D) reconstruction or visualization of forest landscape with 3D tree modeling is more intuitive and can convey more structural and viewshed information about the landscape compared to 2D representations. Moreover, the use of virtual environments built from geographic datasets provide a new capabilities for decision support systems, moving from the traditional exocentric graphic display to a new immersive, egocentric point of view that can mimic field experience. By moving viewpoints through a 3D landscape, this interactive visualization can communicate spatial relationships more clearly.

The motivation of this project is the growing realization that forest landscape visualization (or forest geo-visualization) can serve an important role in GeoDesign, not only as an effective data exploration tool, but also as a communication medium and modeling platform for forest study. We aimed at 1) establishing a data-driven virtual environment workflow capturing forest structure as well as individual tree parameters (*e.g.*, height, species and location), 2) developing visualization of a forest landscape and finally exhibiting it in a 3D space on the Web, and 3) demonstrating and discussing the potential applications of our visualization for forest management or GeoDesign purposes.

In this study, we developed a workflow for the construction of WebGL rendering and interactive visualization of forest landscapes from GIS and forest simulation datasets. We constructed a realistic virtual forest environment for our study site in Virginia. Such Web3D virtual environment, built with X3DOM and HTML5, is capable of fast real-time rendering and walk-through simulation based on two Level-of-Details (LOD) switching techniques, and interactive over mainstream web browsers without a plugin.

2 Related Work

The challenged for the visualization of forest environments include: 2D and 3D data, real geographical information, and user experience and interaction. There are many general-purpose software like 3D Nature and ESRI 3D Analyst that are commonly used for forest visualization. Also a number of forestry-specific application and projects have been developed. Stand Visualization Systems (FVS) is a free tool developed by USDA Forest Service and widely used for rendering symbolic representations of a patch of forest (on Windows) [7]. EnVision – Environment Visualization System, is a non-real-time rendering system for stand and landscape scale images on Windows. It imports USGS DEMs, ground texture, and

---

[*] e-mail:
[†] e-mail:
[‡] e-mail:
[§] e-mail:

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Request permissions from Permissions@acm.org. Copyright is held by the owner/author(s). Publication rights licensed to ACM.

Web3D '17, June 5 - 7, 2017, Brisbane Australia
ACM 978-1-4503-3647-5/17/06...$15.00
DOI: http://dx.doi.org/10.1145/

forest stand information in a Shapefile. osgEarth [17], a third-party terrain creation tool developed and provided by the OpenSceneGraph, is an open source 3D graphics toolkit used for terrain generation and forest visualization [10].

Over the past three decades, advances in computer hardware and software have enabled managers and researchers to visualize the complex structure and dynamics of forested environment using more perceptible and comprehensive computer-aided approaches. Bao *et al* [2] reviewed several techniques to render realistic forests (having thousands of plants, vast amount of geometry) with real-time shadows, such as: Point/Line based rendering, Image based rendering (*e.g.*, billboards), Volume based rendering, Polygon based rendering, Shadows to enhance rendering realism (*e.g.*, Parallel-split shadow mapping). They presented an efficient LOD algorithm to generate multiresolution models, and introduce a new leaf modeling method to have leaf models match leaf textures.

Karjalainen and Tyrväinen [13] presented criteria for evaluating the applicability of visualization methods for forest landscape research purposes, such as realism, simulation of changes in forest landscape, costs of producing visualizations, and the capability of simulation of viewer movement. They envisioned virtual landscape simulators would become easy and flexible means in forest landscape preference research. Their future prospects for the landscape visualization include technical accuracy and data integrity, GIS data integration, and empirical evaluation on the usability of visualization media.

Lim and Honjo [15] described a forest landscape visualization procedure capable of walk-through simulations and its application. They developed a forest landscape visualization system with data of forest stands using Virtual Reality Modeling Language (VRML). They visualized a forest landscape with thousands to tens of thousands of trees, also simulated a variety of forest landscapes and showed how this system can be used to simulate the changes of forest landscapes that occur as a result of natural processes or man-made disturbances such as planting, thinning, and harvesting. Xie *et al* [26] developed a 3D Virtual Reality (VR) system to simulate the evolution of forest ecological landscapes. The system used China's inventory database for selected units (compartment or sub-compartment) based on VRML + ArcGIS Engine. The terrain was modeled from the Digital Elevation Model (DEM). Tree location was calculated using tree density by putting grid points on the subcompartment. Their visualization can display forest changes over time that caused by management activities and disturbances.

The group at Fondazione GraphiTech [16, 19] reviewed virtual forest and 3D environment representation for forestry. They illustrated a web 3D visual simulation system that supports spatial-information based realistic modelling and real-time rendering of forest scenes. They used GeoBrowser 3D which is a customized version of the Cesium JS virtual globe to represent the 3D context, and used HTML5, WebGL, CSS 3D and Canvas element for interactive visualization of the forest model. In all these examples, there are challenges for data preparation and the user interface for delivery: two challenges we address in our work.

3 Methodology

3.1 Data Source

The geospatial dataset used in this project includes: a terrain DEM, high-resolution satellite imagery, generic tree meshes, and photographs of different tree species for textures. The 1/3rd arc-second DEM data was download from United States Geological Survey EarthExplorer data portal (http://earthexplorer.usgs.gov/). It was subsetted to the study area and then used to develop a 3D terrain model. The imagery acquired from national agriculture imagery program (NAIP). It was used as terrain texture and a data source for extraction of tree locations. We used generic tree meshes freely available on *3dvia.com* with customized tree textures to represent different tree species. The texture photograph (leave, branches and barks) for each species were derived from image search engine results that were licensed for non-commercial reuse.

3.2 System Structure

The structure of our geo-visualization system is shown in Figure 1. Raw datasets are collected from DEM, satellite imagery, the field data, tree meshes and textures. In order to align different datasets, namely the terrain data and different type of trees, geo-coordinates of each object are used. Then the different tree models are integrated into the forestation scenario via python script with coordinates. Each point of the terrain model has its geo-coordinates in the same coordinate system as the forestation scenario, thus they are aggregated in the forest geo-visualization and delivered using X3DOM and HTML5.

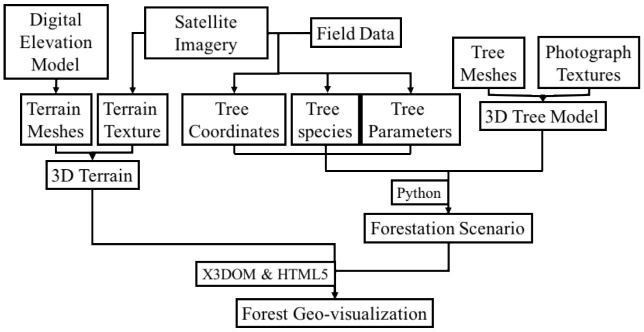


Figure 1: Structure of the proposed geo-visualization system.

3.3 Visualization Techniques

In our implementation, X3DOM is used as the platform to organize and render the aligned 3D models in X3D format. To increase the rendering efficiency and align with the OGC Standard of CityGML, four different LOD models are used. We also provide a simple navigation aid in order to simplify users' interaction with the virtual environments.

3.3.1 X3D and X3DOM

X3D (Extensible 3D Graphics) is the successor to the VRML97 (Virtual Reality Modeling Language). It is an XML-based, open standards ISO file format and run-time architecture to display and communicate 3D scenes and objects. Also it provides a system for storage, retrieval and playback of real time graphics content embedded in applications, all within an open architecture to support a wide array of domains and user scenarios [4, 24]. Acting as a body of certification and advocacy for the open 3D Web, the Web3D Consortium manages the specifications of the X3D file format. The Web3D Consortium is utilizing its broad-based industry support to develop the X3D specification, for communicating 3D on the web, between applications and across distributed networks and web services [25]. X3D has a rich set of componentized features that can be tailored for use in many different applications, among which the geospatial components and implementations *(e.g.,* GeoOrigin, GeoLocation, GeoViewpoint and GeoPositionInterpolator nodes) enable a large class of geospatial visualization applications in standard web browsers [18, 21].

More recently, an open source JavaScript library called X3DOM has been used to display X3D models over the Web using WebGL. X3DOM is an open source framework for loading and rendering declarative Web3D scenes and integrate with the Document Object Model (DOM). It acts as a connector for the HTML5/DOM and the X3D world and content [3]. The visualization is done via standard JavaScript DOM manipulations, such as adding X3DOM elements to the DOM and changing element attributes [20]. Using X3DOM, HTML5, and WebGL, means the application can be run without limitation of OS platforms and run over any browsers that supports WebGL.

A broad spectrum of technologies is available for rendering 3D data formats, which run on the client side (Herman *et al*., 2015). Although a number of geo-visualization tools have been developed to run in Web browsers, many of them require plug-ins, *e.g.*, Google Earth plug-in and popular Adobe Flash plug-in. All plugin-based systems have major drawbacks: they are not pre-installed by default on most system and needs user to deal with installation, security and incompatibility issues [3]. Thus, X3DOM provides a free and convenient platform to create and integrate declarative 3D scenes in Web pages. This property enables user to directly integrate X3D nodes into HTML5 DOM content (Behr *et al*., 2009). In our forest geo-visualization framework, X3D is selected as the 3Ddata format while X3DOM embedded HTML5 is chosen as the delivery platform. The current stable release of the X3DOM distribution (available from www.x3dom.org) which we used in this project is Version 1.7.1.

3.3.2 Level of Detail (LOD) and Cut-Off Function

The concept of an LOD was firstly proposed in 1976 to “structure the environments being rendered” [6]. LOD is aimed at increasing the efficiency of rendering by decreasing the complexity of a 3D object (group of objects) as the observer’s viewpoint moved away from it (them) or according to other metrics. Due to the high complexity and numbers of tree models in forest geo-visualization, LODs is implemented to reduce the workload of the graphics pipeline stages.

Four LODs for each type of tree were created to be rendered at different distance ranges: [*D1*, *D2*, *D3*] (Figure 2). For trees, LOD 4 is designed to implement “billboard”, which consists a picture that faces the user, rotating about the specified axis. LOD 4 will be displayed when the user’s viewpoint is far from the object itself, namely, the distance is larger than$D3$. LOD 1 is the most complex 3D model which is composed of tens of thousands of polygons and will be shown when the distance is less than *D1*. In that case, you are able to have more details of a nearby model than a far one. Both LOD 2 and LOD 3 are 3D models then decrease the number of polygons. Specifically, LOD 2 consists of about three thousand polygons while LOD 3 is made of hundreds of polygons. Moreover, LOD 2 and LOD 3 will be presented when the distance is in the range of [*D1*, *D2*] and [*D2, D3*], respectively.

We created LODs for six species: Beech, Maple, Oak, Pine, Poplar, and Walnut. To maintain the visual appearance of all four LOD models, the textures used for bark and and leaves are the same. By using the LOD technique, rendering computation costs will be balanced with perceived visual fidelity. However, how to determine the appropriate range of distances for an object of a given size remains an open question.

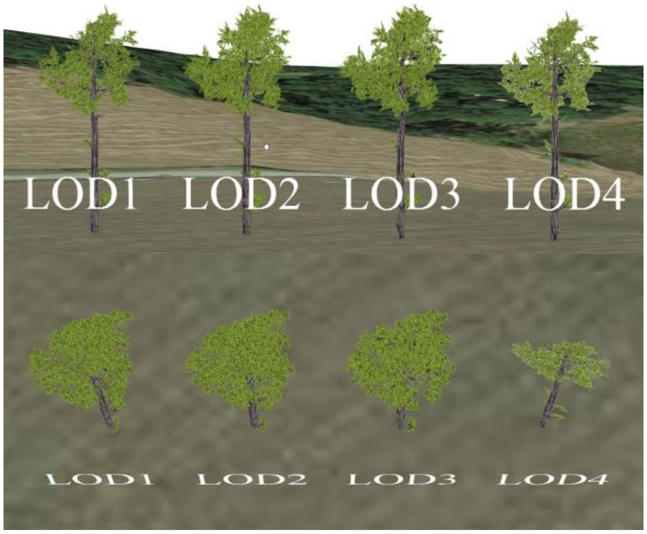


Figure 2: Four LOD levels of a pine tree model

The specific ranges of distance for different LOD levels follows a cut-off function, which represents the relationship between distance and the LOD levels. By implementing different cut-off functions, e.g. an exponential cut-off or a linear cut-off function, the user will be perceive different smoothness or popping as they navigate the virtual environment. We tested two different cut-off functions in a user study, presented below.

3.3.3 Navigation Aid

Navigation plays an important role in the usability of a virtual environment. Since our Web3D scene represents the virtual environment with real geo-coordinates of a real study area, users may need navigation tools such as a compass to show the orientation that shows direction relative to the geographic “cardinal directions”, or “points”. Therefore, we provided a virtual compass to show the user their orientation. In addition, the navigational aid included a distance to target read-out that updated in real-time (Figure 5, bottom). To test the effectiveness of this navigational aid, we conducted an experiment and statistical analysis.

4. Case Study

4.1 Study Area

Our geospatial area is an 836.5 acres (3.4km^2) landscape located at Catawba Valley in Craig County, Virginia. Elevation ranges from

503 to 588 meters (Figure 3). This area also situates the Virginia Tech Catawba Sustainability Center, an outreach and research center that carries sustainable land management projects on silviopasture, agroforestry, and so on.

4.2 Terrain Modeling

3D terrain is the core component of the forest geo-visualization as the other components are built upon the terrain surface. Digital Elevation Models (DEMs) were organized in grid format and used to represent continuous elevation values over the topographic surface; it was referenced to a common datum with the satellite imagery for terrain texture. We explored both statistic and dynamic approaches for 3D terrain generation and rendering from Grid DEM data. The dynamic rendering approach adopted terrain tiling and a hierarchy of terrain-specialized form of Level of Details (*aka*, GeoLOD) technique [22]. In this approach, 3D terrain data and texture are organized in quadtree structure to provide multi-resolution dynamic loading and unloading in real-time according to the distance between viewpoint and terrain tiles. The Geospatial component of X3D describes how to associate real world locations to elements in the X3D world as well as specifying nodes particularly tuned for geospatial applications. With the Geospatial component it is possible to embed geospatial coordinates in the X3D nodes, to support high-precision geospatial modeling, and to handle large multi-resolution terrain data.

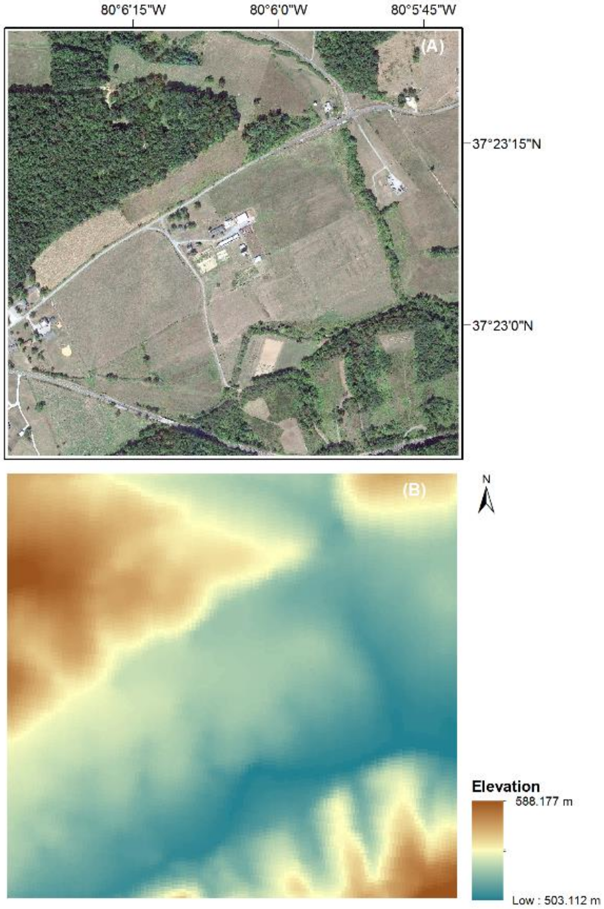
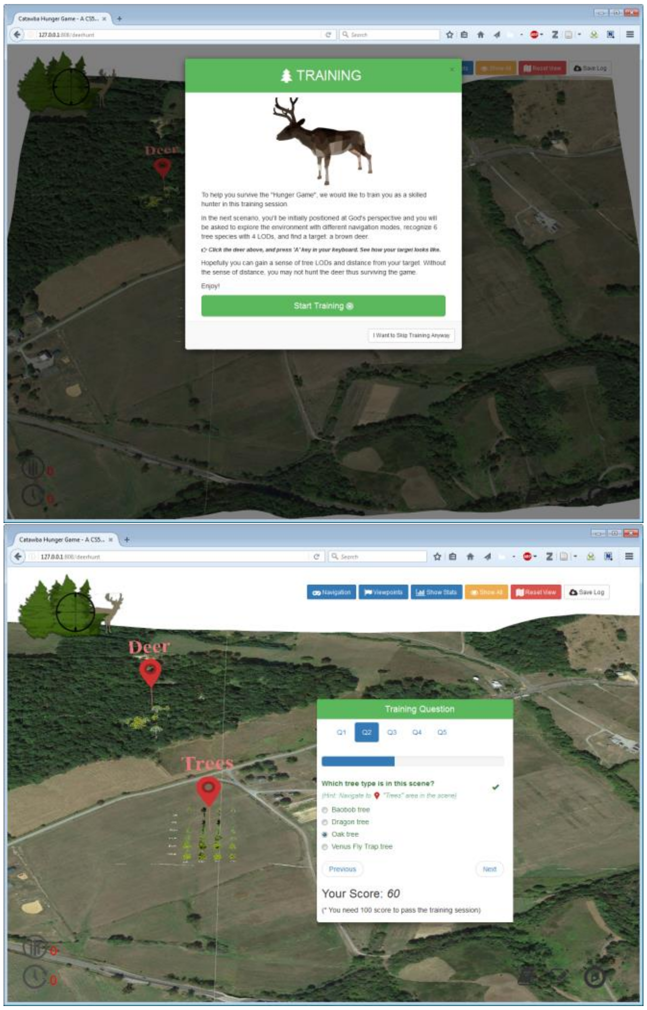


Figure 3: Case study area in Catawba Sustainability Center, Catawba, Virginia: (A) Satellite imagery, (B) Digital Elevation Model

As we have a relatively small location for the case study, we applied a static approach to render the terrain surface by using pre-computed data for visualization to achieve high performance. The 3D terrain mesh was built with local coordinate system UTM Zone 17N NAD 1983 using ArcScene Desktop 10.4, a specialized 3D GIS data visualization application within ArcGIS 3D Analyst (ESRI 2016). The high resolution satellite imagery was overlaid to terrain mesh as a texture to provide a realistic looking of ground surface. The resulting 3D terrain model was exported in VRML format. Then with the Avalon Optimizer (aopt, a free X3D data conversion tool from the InstantReality packages) [12], we transcoded the VRML file to X3D-XML tiles (Figure 3).



To help you survive the "Hunger Game", we would like to train you as a skilled hunter in this training session.

In the next scenario, you'll be initially positioned at God's perspective and you will be asked to explore the environment with different navigation modes, recognize 6 tree species with 4 LODs, and find a target: a brown deer.

Click the deer above, and press 'A' key in your keyboard. See how your target looks like.

Hopefully you can gain a sense of tree LODs and distance from your target. Without the sense of distance, you may not hunt the deer thus surviving the game.

Enjoy!

Start Training

I Want to Skip Training Anyway

Which tree type is in this scene? ✓

(Hint: Navigate to "Trees" area in the scene)

○ Baobob tree
○ Dragon tree
● Oak tree
○ Venus Fly Trap tree

Your Score: *60*

(* You need 100 score to pass the training session)

Figure 4: Terrain overview and training screenshots

4.3 Tree Modeling

Forest vegetation includes trees, bushes and herbal plants. In this project, we focused on 3D modeling of trees while other vegetation cover were represented by the ground texture from satellite imagery. Generally, two emphases are taken into account in visualizing forest virtual environment: individual tree visualization and forest scene visualization [11]. There are two pathways in modeling individual tree, the analytical and visual pathways. The former aims at accurate modeling of biological attributes of tree species (*e.g.*, height, diameter at breast height, canopy diameter, *etc*.) in serving statistical analyses. This often involves botany-based approaches in plant simulation and abstract tree modeling. On the other hand, the visual pathway lays emphasis on efficient and effective rendering of 3D objects to present realistic and impressive results.

We took the visual pathway in developing our tree models. To visualize single tree, a total of six local tree species, oak, pine, maple, poplar, beech and walnut, which are commonly found in Virginia were collected and modelled in 3D. In creating individual tree renderings, we developed 3D models based on the generic tree meshes and applied textures to each tree model from real photographs of bark, leaves and branches. We also incorporated some aspects related to analytical pathway. Aside from tree species, we attempted modeling of individual tree attributes (which only include tree height at this point) by manipulating model parameters of each tree model.

Realistic and effective rendering of tree models is critical for forest geo-visualization. As a forest is often complicated and extensive, the rendering of a forest scene is often time-consuming and difficult to generate at run-time. In creating and rendering a forest scene, tree LODs and "billboard" modeling was employed to get both fast and visually impressive rendering. Tree LODs were applied to reduce the complexity of models and to optimize the scene rendering with a moving viewpoint. For each tree species, we developed four LODs. The LOD1, LOD2 and LOD 3 are 3D geometric tree models with gradually simplified complexity. We created an X3D-XML file for each level (Table 1).

Table 1: Tree LOD model sizes

Species	Model Size (MB) (LOD 1,2,3,4)	Texture Size (MB)
Beech	1.1, .391, .255, .001	.906
Maple	3.4, 1, .488, .001	.89
Oak	.628, .1, .05, .001	.63
Pine	4.4, .988, .497, .001	.65
Poplar	1.4, .2, .09, .001	1.33
Walnut	1, .47, .27, .001	1.1

LOD 4 is only a tree image mapped onto a simple 2D rectangular vertical face (billboard). The background of the texture image is transparent. We use this billboard technique to create a high degree of realism for the tree when viewed from a far distance. When a viewer moves from an overlook of the forest (or at a far distance to a tree) to the walk-through in the forest (or at a near distance to a tree), the total number of trees to be rendered within the current field of view is reduced while the complexity of tree models is increased. The billboard tree LOD model switches to a geometric and textured LOD model to keep a consistent perception of the forest scene (Figure 2).

The 3D coordinates of trees (longitude, latitude and altitude) were mapped from satellite imagery in GIS software. We extracted tree cover with a spectral vegetation index -- Normalized Difference Vegetation Index (NDVI). NDVI is a numerical indicator that uses the visible and near-infrared bands of the electromagnetic spectrum, and is adopted to analyze remote sensing measurements and assess whether the target being observed contains live green vegetation or not. The forest patches were delineated in Image Analysis module of ArcMap Desktop 10.4 based on a threshold of NDVI value.

Although we can further distinguish tree canopy to get individual tree locations based on object-based image classification and segmentation, we adopted a simple approach that generates random trees within the forest patches. We converted the forest patch areas into polygons in ArcMap, defined the maximum and minimum distance among points (*i.e*., range of tree densities), and then create random points within each polygon to get the estimated tree coordinates for our study area. This lookup table for tree locations was used to "plant" each tree model on top of the terrain model, as well as to integrate tree modeling with landscape modeling (terrain and other anthropogenic objects).

5. Empirical Evaluation

We undertook a user study to evaluate our LOD trees and navigation aid to increase the usability and accessibility of our forest geo-visualization, In order to increase participants' engagement and motivation to explore the space and examine the landscape, we applied a "gamification" concept in application design by integrating a "deer hunting" element into the visualization application. This also provided a context for participants to navigate the environment and search out targets in the woods. Specifically, we built several forest planting scenarios with different tree distributions and different deer positions (Figure 7). We conducted an experiment involving 15 participants to play the game while we gathered usability metrics.

5.1 Hypotheses

In this study, we set out with two different hypotheses regarding our forest geo-visualization user interface. These are as follows:

- Hypothesis 1: the Navigation aid will significantly improve users' performance.
- Hypothesis 2: the exponential cut-off function will provide a smoother transition switching between Levels of Detail

5.2 Experimental Setup

We set to evaluate the functionality and performance implications of our navigation aid and two types of cut-off functions. We considered the navigation aid and the different cut-off as independent variables. We implemented a full factorial design with four conditions and two trials in each condition. Two levels were selected for each factor: navigation aid (on, off) and cut-off function (exponential cut-off, linear cut-off). Each subject was shown in total eight trials in the same orders that follow the experimental setting shown in Table 2.

We random generated eight different planting scenarios (maps of tree locations) and write to a csv file; then use a python script to build the resulting X3D file for the forest scene. The average number of trees across all trials were 1486 (stdev = 322). To reflect the local vegetation structure and composition, the proportions of tree species in our planting scenarios were approximately 50% Pine, 13% Walnut and Maple, 10% Poplar and Oak, and 4% Beech.

The task for each participant was to find and "shoot" (by mouse click) a white tail deer hidden in the forest. The location of the deer was set to a different place in each of the eight scenarios. The exponential cut-off function were implemented in scenarios 1~4 while the linear cutoff function was used in scenarios 5~8. The LOD cut-off functions are shown in Figure 5. By applying the models, the range of distances for our unit tree model can be obtained as: 1) exponential: $[44.3, 277.9, 813.4]$, 2) linear: $[50, 525, 1000]$. The difference between the two cut-off functions for our scale lies at just one distance. The navigation aid (the compass and distance display in the bottom right of the screen) presents the orientation of the user, and was displayed in trials 1,3,5,7 and hidden in trials 2,4,6,8.

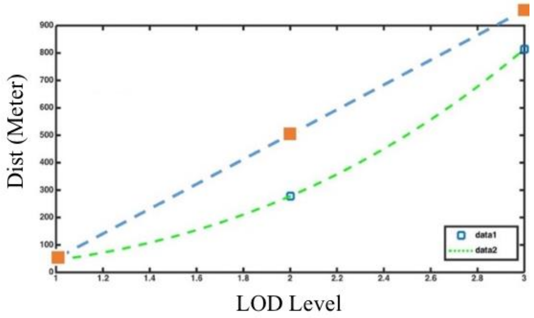
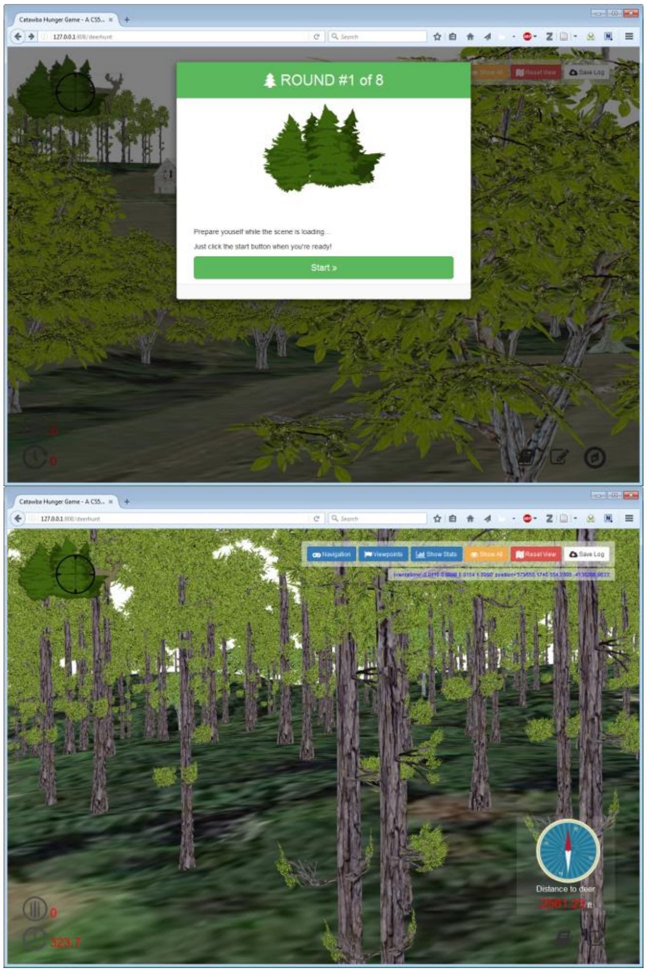


Figure 5: Exponential cut-off and linear cut-off functions.

The user study took place at 3050 lab in Torgersen Hall at Virginia Tech. Total of 15 participants were recruited among current undergraduate and graduate students with various educational background and levels of computer skills. Before the study, the participant was asked to fill out questionnaires related to experiences with computer skills and virtual geographic environment (such as Google Earth or Bing Map 3D). Each participant attended a session lasted approximately 25 minutes. All user tasks were performed on a desktop computer (specs: Inter Core i7-3770K CPU @ 3.50GHz, 32GB RAM, Windows 10 operation system) attached with a 27-inch monitor, mouse and keyboard (Figure 6). The brightness and contrast of the monitor was set at a comfortable level for the participant. The web3D virtual environment was delivered via a Google Chrome web browser. The entire session of the experiment includes: 1) *training*: get familiar with application tutorial and pass through a training application, 2) *gaming*: perform tasks in the deer hunting game, and 3) *feedback*: rate user experiences at the end of each task.

We designed a user training stage to help every participant get familiar with navigation interaction in our virtual forest environment, recognize the deer target and tree LODs, and gain a sense of distance in virtual environment. There were five step-by-step training questions to direct the participant to navigate the virtual forest landscape and accomplish the training. The questions were about the buttons of this web3D application interface, tree species, and concept of LOD. Initially, the participant's viewpoint was positioned at a point of 1,000 feet above ground, and then he/she was directed to "fly" to the ground and "walk" to a tagged area by switching and practicing different navigate modes in the virtual environment. A navigation aid is optional to use to learn about distance to the target and orientation in the scene. The participants must explore both "Trees" and "Deer" areas to get enough details in order to answer all training questions (Figure 4). There is a short break (1~2 minutes) between the training stage and gaming stage.

Figure 6: The physical condition of experiment



Prepare youself while the scene is loading...

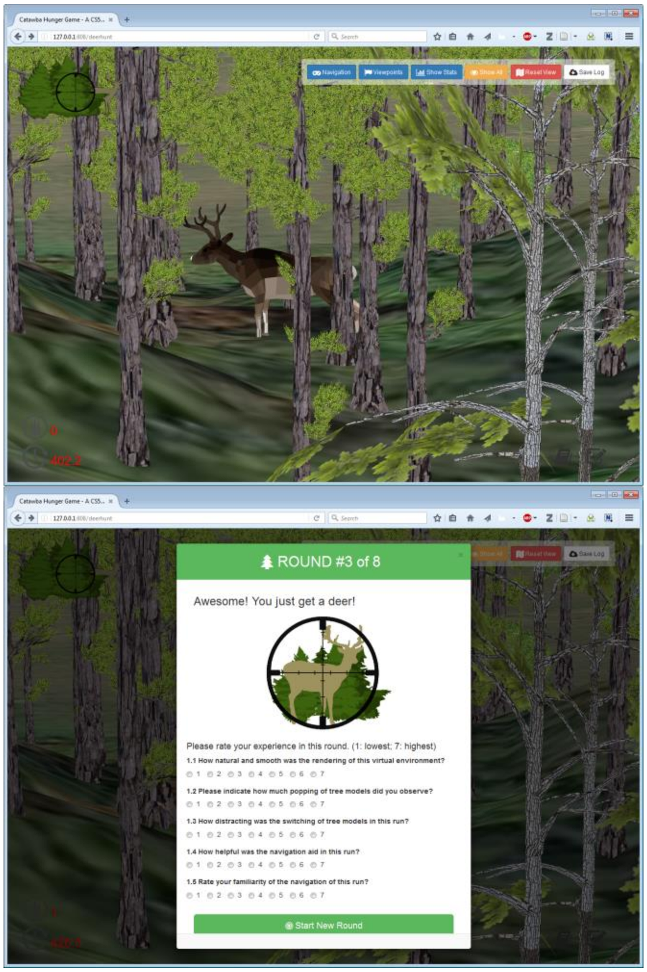
Just click the start button when you're ready!

Figure 7: Deer hunting game based on forest geo-visualization

In the gaming and feedback stages, each participant performed eight tasks with two different LOD cutoff function settings and different availabilities of navigation aid. In each task, the participant was allowed to navigate freely in the virtual environment to find a deer. There is a minimum distance required between the hunter's location and the deer, so the participant must get close enough to be able to hunt by mouse click after spotted the target. After each run (a successful hunt in each scenario), he/she was asked to rate user experience on a scale of 1 to 7 based on five questions (Figure 8 bottom):

i. How natural and smooth was the rendering of this virtual environment?
ii. Please indicate how much popping of tree models you observed.
iii. How distracting was the switching of tree models in this run?
iv. How helpful was the navigation aid in this run?
v. Rate your familiarity of the navigation of this run.

Meanwhile, the elapsed time of the participant was recorded for each scenario. The performance data and the questionnaire scores were used in the statistical tests.



ROUND #3 of 8

Awesome! You just get a deer!

Please rate your experience in this round. (1: lowest; 7: highest)

1.1 How natural and smooth was the rendering of this virtual environment?
○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

1.2 Please indicate how much popping of tree models did you observe?
○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

1.3 How distracting was the switching of tree models in this run?
○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

1.4 How helpful was the navigation aid in this run?
○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

1.5 Rate your familiarity of the navigation of this run?
○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7

Start New Round

Figure 8: The deer target in the game and user experience rating questionnaires

Table 2: Experimental setting

	Navigation Aid On	Navigation Aid Off
Exponential Cut-Off	Scenario 1, 3	Scenario 2, 4
Linear Cut-Off	Scenario 5, 7	Scenario 6, 8

5.3 Results and Analysis

The effects of our navigation aid across all scenarios was analyzed using time elapsed for all participants. The mean time elapsed with the navigation aid was 130.26 seconds with a standard deviation of 127.44. The mean time elapsed without the navigation aid was 178.94 seconds with a standard deviation of 194.44. Thus, the navigation aid seems to provide a significant advantage in terms of performance times and the consistency of performance.

Participants' smoothness ratings were analyzed to reflect the effects of cut-off functions. Since participants' familiarity with the virtual environment was also a variant which impact the performance of deer hunting games. Therefore, we applied a standardization to centralize the subjects' smoothness ratings to have zero mean, and scale them to ensure the unit variance[5].

To test our hypotheses listed in Section 5.1, we performed a paired t-test to find the difference between the two levels of each factor. Afterwards, box plots and paired t-tests were conducted and the result is shown in Figure 8.

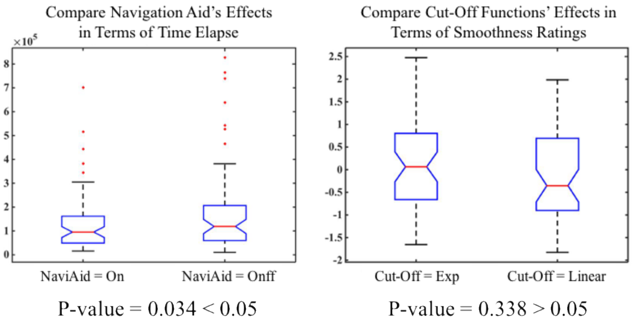


Figure 8: Paired t-test results and box plot of standardized time elapse and smoothness ratings.

According to the results, for the first hypothesis, the null hypothesis can be rejected; thus, the navigation aid has significantly increased the user's performance. This result makes sense because the navigation aid provides both the orientation of the user and the distance between the user and the target object. Without the navigation aid, the user cannot easily locate himself/herself relative to the target thus spends much more time self-orientating instead of searching for the deer.

However, for the second hypothesis, there's no statistical confidence to reject the null hypothesis, which means that in our study, the exponential cut-off function and the linear cut-off have no significantly different effects from users' perspective. This result also makes sense because the appearance of four LOD models is relatively consistent and so the switching of LODs is not easily noticed. In a virtual environment, users have a hard time judging distance and speed unless they are walking.

6. Conclusion and Future Work

Interactive 3D forest visualization can facilitate research into forest structure, spatial patterns, dynamics, and the growth of individual tree across the landscape. While a number of forest visualization tools have been developed, most of them are standalone software or web-based applications that require specific plug-ins. Although a few web-based visualization tools that simply require a web browser have been developed, the tradeoff of low fidelity and acceptable rendering computation costs has not been well balanced. We demonstrate a lightweight pipeline and as a web-based forest geo-visualization system and implemented it to a study area via X3DOM and HTML5. We used LOD techniques to decrease the rendering computation costs while maintaining high fidelity. We also observe that our navigation aid increases the usability of the visualization tool.

In our study of a Web3D interface for forest geo-visualization, we explored the performance of our navigation aid and two types of LOD cut-off functions on tree models. According to the paired t-test results, participants' performance was significantly improved with the navigational aid, while the exponential cut-off versus linear cut-off functions showed no significant differences from participants' perspective. Although we observed no significant performance benefits on either of the two range-based LOD switching functions for individual trees, this could be a consequence of our 3D models and texturing: we developed the models for each level by hand and so made great care to keep consistent visual appearance.

Future prospects for expanding this work target on more detailed forest visualization covering a larger geographic area, dynamic terrain generation and landscape models with GeoLOD [20, 21], and live database queries from Web3D WxS services[14]. All 3D models in the forest geo-visualization should be delivered using a single geographic coordinate system within a Web 3D service. Finally, we also propose that it is worth considering the forest itself as one whole object, and explore other LOD models optimized for forest visualization and other X3D-based LOD switching techniques on a forest stand level.

References

[1] 3D forest reconstruction to improve environmental monitoring: 2015. *http://www.npl.co.uk/news/3d-forest-reconstruction-to-improve-environmental-monitoring*. Accessed: 2016-05-08.

[2] Bao, G., Meng, W., Li, H., Liu, J. and Zhang, X. 2011. Hardware instancing for real-time realistic forest rendering. *SA '11 SIGGRAPH Asia 2011 Sketches* (Hong Kong SAR, China, 2011), 1.

[3] Behr, J., Eschler, P., Jung, Y. and Zöllner, M. 2009. X3DOM: a DOM-based HTML5/X3D integration model. *Web3D '09 Proceedings of the 14th International Conference on 3D Web Technology* (2009), 127.

[4] Brutzman, D. and Daly, L. 2010. *X3D: Extensible 3D Graphics for Web Authors*. Morgan Kaufmann.

[5] Cheadle, C., Vawter, M.P., Freed, W.J. and Becker, K.G. 2003. Analysis of Microarray Data Using Z Score Transformation. *The Journal of Molecular Diagnostics*. 5, 2 (May 2003), 73–81.

[6] Clark, J.H. 1976. Hierarchical geometric models for visible surface algorithms. *Communications of the ACM*. 19, 10 (Oct. 1976), 547–554.

[7] Dixon, G.E. 2002. *Essential FVS: A user's guide to the Forest Vegetation Simulator*. U. S. Department of Agriculture, Forest Service, Forest Management Service Center.

[8] Dunbar, M.D., Moskal, L.M. and Jakubauskas, M.E. 2004. 3D Visualization for the Analysis of Forest Cover Change. *Geocarto International*. 19, 2 (Jun. 2004), 103–112.

[9] Falcão, A.O., Santos, M.P. dos and Borges, J.G. 2006. A real-time visualization tool for forest ecosystem management decision support. *Computers and Electronics in Agriculture*. 53, 1 (Aug. 2006), 3–12.

[10] Huang, H., Tang, L., Li, J. and Chen, C. 2012. Simulation and visualization of forest fire growth in an integrated 3D virtual geographical environment - a preliminary study. (Hong Kong SAR, China, Jun. 2012), 1–6.

[11] Huang, H., Tang, L., Li, J. and Chen, C. 2012. Simulation and visualization of forest fire growth in an integrated 3D virtual geographical environment - a preliminary study. (Hong Kong SAR, China, Jun. 2012), 1–6.

[12] InstantReality 2016. *instantreality Framework*. Fraunhofer Institute for Computer Graphics Research IGD.

[13] Karjalainen, E. and Tyrväinen, L. 2002. Visualization in forest landscape preference research: a Finnish perspective. *Landscape and Urban Planning*. 59, 1 (Mar. 2002), 13–28.

[14] Kim, J.-S., Polys, N. and Sforza, P. 2015. Preparing and evaluating geospatial data models using X3D encodings for web 3D geovisualization services. *Proceedings of the 20th International Conference on 3D Web Technology* (2015), 55–63.

[15] Lim, E.-M. and Honjo, T. 2003. Three-dimensional visualization forest of landscapes by VRML. *Landscape and Urban Planning*. 63, 3 (Apr. 2003), 175–186.

[16] Panizzoni, G., Magliocchetti, D., Prandi, F. and De Amicis, R. 2015. Interactive Virtual Planning Tools for Sustainable Forest Production in Mountain Areas. *HCI International 2015 - Posters' Extended Abstracts*. C. Stephanidis, ed. Springer International Publishing. 220–225.

[17] Pelican Mapping *osgEarth*.

[18] Plesch, A. and McCann, M. 2015. The X3D geospatial component: X3DOM implementation of GeoOrigin, GeoLocation, GeoViewpoint, and GeoPositionInterpolator nodes. *Proceedings of the 20th International Conference on 3D Web Technology* (2015), 31–37.

[19] Prandi, F., Panizzoni, G., Magliocchetti, D., Devigili, F. and De Amicis, R. 2015. WebGL virtual globe for efficient forest production planning in mountainous area. (2015), 143–151.

[20] Sharakhov, N., Polys, N. and Sforza, P. 2013. GeoSpy: a Web3D platform for geospatial visualization. *Proceedings of the 1st ACM SIGSPATIAL International Workshop on MapInteraction* (Orlando, Florida, 2013), 30–35.

[21] Sharakhov, N., Polys, N. and Sforza, P. 2013. SpeedSpy: a mobile Web3D platform for visualizing broadband data. *Proceedings of the 18th International Conference on 3D Web Technology* (San Sebastian, Spain, 2013), 208–208.

[22] Tourtelotte, D.R. 2010. *X3D-Earth: Full Globe Coverage Utilizing Multiple Datasets*. Naval Postgraduate School.

[23] Wang, X., Song, B., Chen, J., Zheng, D. and Crow, T.R. 2006. Visualizing forest landscapes using public data sources. *Landscape and Urban Planning*. 75, 1–2 (Feb. 2006), 111–124.

[24] Web3D Consortium 2015. *X3D: The real-time 3D solution for the world-wide web*.

[25] What is X3D: 2016. *http://www.web3d.org/x3d/what-x3d*. Accessed: 2016-05-09.

[26] Xie, X., Liu, Z., Su, D., Dai, L., Wang, X. and Qi, G. 2010. Forest Landscape 3-D Visualization from China's National Forest Inventory Spatial Database. (Chengdu, 2010), 1–4.