Design of High-precision Island WebGIS Based on Cesium

Qingjun Zhang
Jiangsu Province's Key Lab of Image Procession and Image Communications, Nanjing University of Posts and Telecommunications, Nanjing, China
1219012711@njupt.edu.cn

Dong Hu
Jiangsu Province's Key Lab of Image Procession and Image Communications, Nanjing University of Posts and Telecommunications, Nanjing, China
hud@njupt.edu.cn

Qiang Lin
Jiangsu Province's Key Lab of Image Procession and Image Communications, Nanjing University of Posts and Telecommunications, Nanjing, China
1218012401@njupt.edu.cn

ABSTRACT

In this paper, the design of a high-precision WebGIS system for island application is presented. The system is developed based on Cesium to support 2D, 2.5D and 3D map capabilities, and provide networked comprehensive geographic information service. Concerning the practical requirements for complicated configuration of island surface, improved methods for interpolation correction, data structure optimization, visible analysis and island path planning are introduced to improve system accuracy and performance. The system adopts B/S architecture and modular development ideas for easier access and further updates. The main functional modules of the island WebGIS provide basic operations, including multi-dimensional scene browsing, base map switching, multi-control operation, layer plotting, contour line, intervisibility and terrain factors measurement. Besides, the characteristic functions of key techniques such as profile analysis, viewshed analysis, and island path planning are implemented. The test examples show that the overall functional performance of the system is satisfactory for island 3D GIS service.

CCS CONCEPTS

• **Information systems**; • **Information systems applications**; • **Spatial-temporal systems**; • **Geographic information systems**;

KEYWORDS

WebGIS, Cesium, Island, Geographic information system

ACM Reference Format:
Qingjun Zhang, Dong Hu, and Qiang Lin. 2021. Design of High-precision Island WebGIS Based on Cesium. In *The 5th International Conference on Computer Science and Application Engineering (CSAE 2021), October 19–21, 2021, Sanya, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3487075.3487146

CSAE 2021, October 19–21, 2021, Sanya, China

ACM ISBN 978-1-4503-8985-3/21/10…$15.00
https://doi.org/10.1145/3487075.3487146

1 INTRODUCTION

WebGIS is the combined product of Internet and GIS, which possesses good characteristics with flexibility, expansibility and cross-platform application. A number of the relative geographic information development tools have been derived, such as MapInfo-Proserver, ArcGIS API for JavaScript, SuperMap, GoogleEarth, etc. Besides, some scholars also have focused on different application scenarios on the WebGIS platform. Guiseppe Modica et al. [1] proposed a WebGIS platform completely open and based on OGC standards and services and allowing to perform geospatial analyses by users. In recent years, Cesium, as a JavaScript open source WebGIS product for 3D virtual Earth, has the characteristics of cross-platform, lightweight and plugin-free, providing technical support for users to intuitively display and efficiently process geographic information data. Cesium has also become particularly important since the popular Google Earth plugin was obsolete. So far, the research and application development of 3D WebGIS based on Cesium have been reported successively. For examples, Sun [2] described the construction, loading and display of 3D terrain based on the visualizing library of Cesium.js; He et al. [3] built power GIS system based on Cesium and 3D CityDB, and carried out panoramic visualization of power data; SM Murshed et al. [4] developed an application program for intelligent city analysis visualization based on Cesium. Islands are important land resources. GIS has been applied more and more widely in islands, but most are basic studies such as island name classification, soil and engineering geology analysis. For example, Nistor et al. [5] analyzed soil characteristics and distribution of islands based on GIS. Yang et al. [6] evaluated the suitability of the island engineering construction geology based on GIS spatial analysis function and geological factors. Due to the diversity and complexity of islands, the basic 2D research can no longer satisfy a deeper understanding of the 3D features of islands. Therefore, it is of great significance to build an informative and visual island GIS to provide efficient, intuitive and high-precision island management and analysis tools for the development, exploration and protection of islands. So far, powerful 3D GIS is developed based on C/S architecture, but it greatly limits the development, upgrade and management of the system. Based on the above requirements and deficiencies, this paper uses Cesium framework for secondary development and optimizes some core function algorithms from the system bottom, aiming to build a smooth and high-precision island WebGIS and provide networked comprehensive geographic information services. The paper introduces the technical points of the system design, including system framework, core function organization, and main technical support used to improve accuracy and efficiency. Finally, the experiments

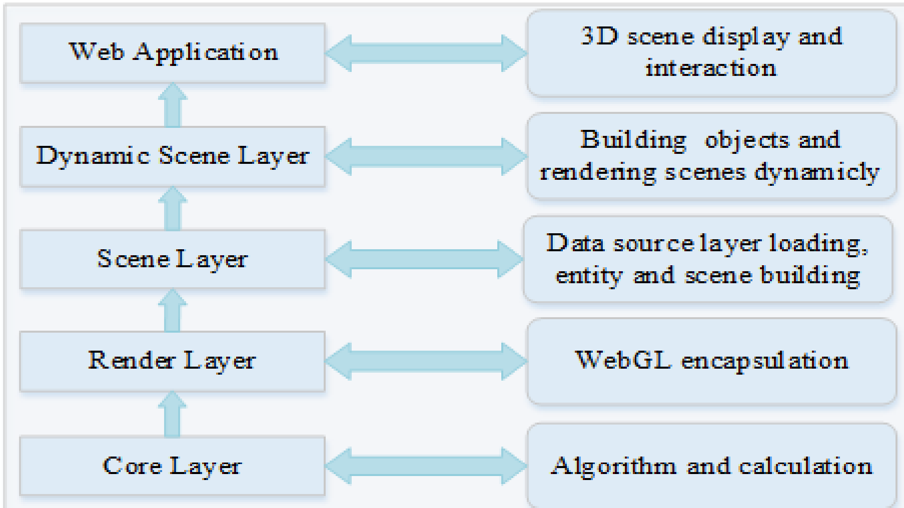

Figure 1: Cesium Framework Structure.

show the basic operations including scene browsing, roaming, plotting, terrain measurement and analysis. Besides, some characteristic technologies are presented for the unique attributes of the island, such as visual analysis, mountain path planning and inundation analysis.

The rest of the paper is organized as follows: Section 2 briefly introduces the Cesium framework and its key classes, as well as the technical requirements in development; Section 3 introduces the system architecture, the organization of key modules, as well as the main technical support and characteristic modules; The functional effects of the system are presented in Section 4, followed by a conclusion in Section 5.

2 CESIUM FRAMEWORK AND KEY CLASSES

Cesium is developed based on HTML5 and the underlying framework of WebGL. It can load massive 3D models and global remote sensing images and topographic data more smoothly, and realize the integration of 2D and 3D. The Cesium framework structure and functions of each layer is shown in Figure 1. From the core layer to the application layer, Cesium takes full advantage of hardware acceleration, combined with API and built-in high-precision algorithms to render and visualize. The five-level modules are closely connected, providing a multi-functional and easy-to-use interface, which is convenient for secondary development.

Cesium provides many API classes, among which viewer class, also the core class, is the main development entrance to initialize 3D virtual earth. When the scene is initialized, it is necessary to create a new visual window with the Viewer and nest the main functional classes in it. In the Viewer, the ImageProvider class and TerrainProvider class are used to load images and terrain data offline or online, which is to realize a real 3D scene. The Widgets class can add page controls, such as Animation, InfoBox, baseLayerPicker and so on. The Camera class is used to control the view of the scene such as rotate, zoom, pan, flyto, etc. Moreover, the Cesium has two methods for adding entities, namely the Entity and Primitive. Besides, the Cesium3DTileset class is also a necessary class to process and load 3D tiles data. As a complement, some Event classes can handle interaction with Camera for scene browsing and roaming. The above key classes are mainly used in development and realization of our system. Besides, Cesium provides more detailed functional classes for the development.

3 SYSTEM FRAMEWORK DESIGN AND CORE TECHNOLOGY

The design of this system uses advanced WebGIS framework and core technology to achieve efficient management and utilization of island geographic information data, and highly visualized expression. In the following, the discussion of the overall system architecture is presented. Then, the technical details are described to meet the needs of users for the real scenes of the island, terrain analysis and spatial analysis.

3.1 The Overall System Framework

The island WebGIS system is designed with B/S architecture, and it can be operated smoothly and visually to obtain geographic information services through a Web browser. The overall system architecture is shown in Figure 2

The system is generally divided into the server end and the browser end. The service end deployment adopts the Node.js running environment, which receives different HTTP requests from clients and processes data through the listening port of the server. At the browser end, Cesium framework is used to construct 3D earth scene to realize visual interaction between users and WebGIS system. Both request and respond to data through the Ajax technology. From the function level perspective, the system mainly consists of the data layer, the business layer and the user layer. The user layer utilizes the improved algorithm in Cesium core layer and API development classes to carry out secondary development. Then, the

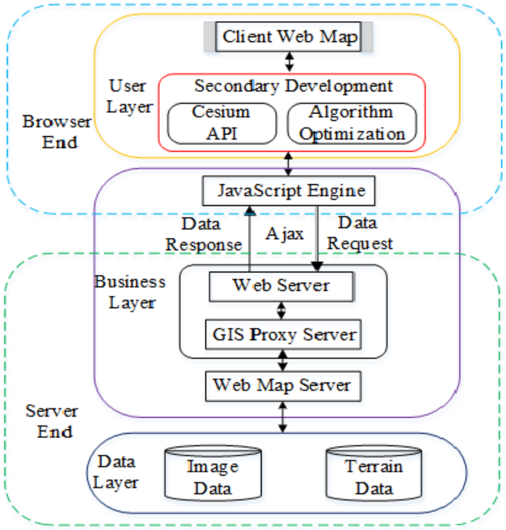

Figure 2: The Overall System Architecture

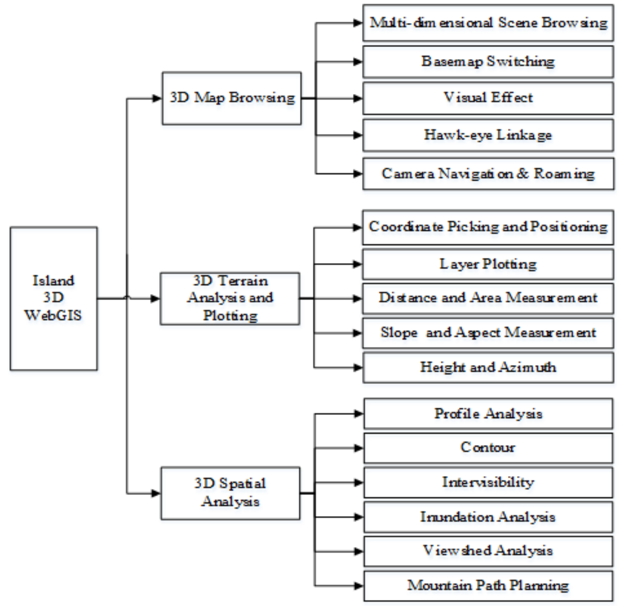

Figure 3: System Function Modules.

geographic information data can be visualized efficiently through WebGL technology to realize the presentation of functional scenes. The business layer mainly deals with data request and response, and connects with the database. The data layer is used to store geographic information data.

3.2 Main System Function Modules

For the special island environment and the needs of development, the system functions are mainly divided into three large modules in the modular idea, and their sub-functions are further subdivided, as shown in Figure 3

3.2.1 3D Map Browsing. Initialize 3D scene to intuitively visualize the topography of the island. Scene experience in different modes can be realized by switching the base map, adding visual effects, hawk-eye linkage or mouse multi-control operation. Besides, users can roam in the air and along the ground to achieve real VR experience.

3.2.2 3D Terrain Analysis and Plotting. This module provides basic terrain factor calculation and layer plotting functions for terrain analysis. While browsing the island, users can mark on the study area and measure the topographic data, such as distance, area, angle, slope, aspect, and display the measurement results on the view in time.

3.2.3 3D Spatial Analysis. This module includes the basic functions such as profile, contour line, intervisibility, as well as the island characteristic functions such as viewshed analysis, island path planning and inundation analysis. This part fully combines the island special geographical environment and terrain attributes, optimizes the algorithm to realize the high-precision and efficient analysis.

3.3 Supporting Technology and Technical Characteristics

This paper mainly achieves the high efficiency of the system from two aspects. One is to improve the system efficiency with faster loading speed, stronger rendering ability and real smooth scene; the other is to achieve high precision expression, that is to achieve close to the real 3D scene, more accurate measurement and analysis results. To achieve the purpose above, several supporting technologies applied in system development are presented in the following parts.

3.3.1 Data Hierarchical Loading. The tile pyramid model [7] is a multi-resolution hierarchical structure, which can dynamically load and display the scene data through layering and partitioning. This island system realizes the large scene application by using offline Digital Elevation Model (DEM) and image data with the characteristics of huge amounts. To reduce the memory occupation and improve the rendering speed of page scenes, this paper adopts the form of quadtree to build the pyramid for data loading schedule. The overall data is graded and sliced through the quadtree structure. The tile data files are saved according to the scene scaling ratio levels and the detailed information is recorded in the JSON file, as shown in Figure 4. During rendering, Cesium loads tile data by the index of tile data in the JSON file. It can dynamically change the splitting and merging states of terrain quadtree nodes according to the distance of view point and local terrain roughness, and adjust the display levels of terrain in time.

3.3.2 Elevation Interpolation and Correction. DEM is the data source of terrain and spatial analysis. The accuracy of elevation data is greatly affected by the complex topography of the island, which leads to data loss and large analysis error. Therefore, when the elevation of multiple unknown sampling points is calculated,

Figure 4: Data Hierarchical Loading. (a)The Quadtree Structure. (b) The Tile Data Files.

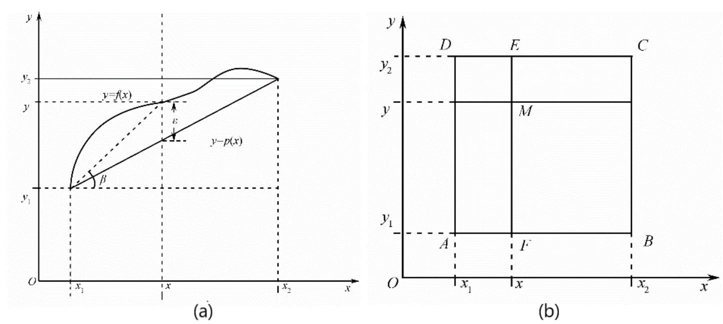

Figure 5: Elevation Interpolation. (a) The Terrain Profile. (b) Bilinear Interpolation.

the elevation interpolation should be implemented combined with the special topographic factors of the island. The commonly used bilinear interpolation [8] is suitable for the interpolation of grid DEM with high efficiency. However, it fails to consider the topographic features of the grid when describing the actual surface of the islands with mountainous areas, hills and coasts, resulting in the loss of interpolation accuracy. Since DEM accuracy is strongly correlated with the slope and shape of terrain [9], in view of the accuracy and efficiency, we apply bilinear interpolation method with slope factor correction to accurately interpolate unknown points. The errors of traditional bilinear interpolation are mainly derived from the approximate description of the actual terrain $y = f(x)$ using linear function $y = p(x)$, as shown in Figure 5(a). The elevation error is

$$\varepsilon = f(x) - p(x) \tag{1}$$

The island topography is complex, and the slope$\beta$and shape ρ vary significantly. They are shown in Equation (2).

$$\begin{cases} \rho = H_{i,j} - \frac{1}{n}\sum_{i-1}^{n} H_i \\ \beta = \arctan\sqrt{f_x^2 + f_y^2 \times \frac{180}{\pi}} \end{cases} \tag{2}$$

Where $H_{i,j}$ is the center elevation of 3×3 neighborhood window, H_i is the elevation of a grid place in window, and n is the number of the window grid points f_x and f_y are elevation change rates in the x and y directions respectively, which are calculated by the difference algorithm of numerical analysis method in the regular grid.

In Figure 5(a), the relationship between slope and elevation error is analyzed from a geometric perspective, as shown in Equation (3).

$$\tan\beta = \frac{\varepsilon/\alpha_y}{\alpha_x(x_2 - x_1)} \tag{3}$$

Where α is the scaling coefficient, and $0 \le \alpha \le 1$, which changes dynamically with different slopes. β is the slope at (x_1, y_1) $\lambda = \alpha_y\alpha_x(x_2 - x_1)$, and it is defined as the correction factor. λ can correct the error by judging the slope and shape of each interpolation point.

In Figure 5(b), with the approximate slope of the interpolation point, the elevation of the target point M is calculated by combining the correction factor λ and the elevation of the surrounding known points, as shown in Equation (4).

$$f_M = \begin{cases} f_F + (y - y_1)(f_E - f_F) + \lambda\tan\beta_M, \rho_M > 0 \\ f_F + (y - y_1)(f_E - f_F), \rho_M = 0 \\ f_F + (y - y_1)(f_E - f_F) - \lambda\tan\beta_M, \rho_M < 0 \end{cases} \tag{4}$$

Where f_E and f_F can be obtained from the four elevation points around.

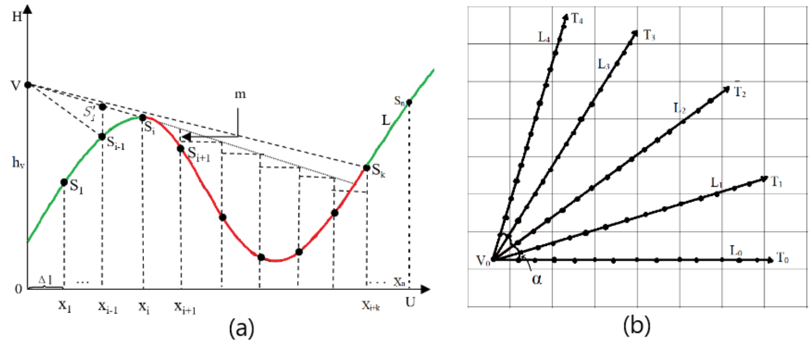

Figure 6: Visual Analysis. (a) Intervisibility Analysis Based on Visual Alternation. (b) LOS Scanning in Viewshed.

3.3.3 Visual Analysis Based on Visual Alternation. With the inspiration of traditional algorithm [10], this paper proposes a visual analysis algorithm based on visual alternation, which fully combines the advantages of Cesium and visual reusability. In the intervisibility, the line of sight (LOS)VT cuts the terrain XYvertically to obtain a continuous profile curve L. Generally, the visible and the invisible parts appear strictly alternately on curve$L$, as shown in Figure 6 (a). When judging whether the current point S_i is visible, if the previous S_{i-1} is visible, then its height $H(S_i')$ on the LOS VS_i can be shown in Equation (5).

$$H(S_i') = \frac{|H(V) - H(S_i)|}{n} \times (n-1) + H(V) \tag{5}$$

Where n is the sampling ordinal number of S_i and $H(V)$ is the height of the viewpoint V. If $H(S_i') > H(S_i)$, the current point S_i is visible, otherwise it is invisible.

If S_{i-1} is invisible, it is necessary to backtrack to the nearest previous visible point S_k and recalculate $H(S_i')$ based on the current LOS, as shown in Equation (6) and (7).

$$m = \frac{H(S_k) - H(V)}{k} \tag{6}$$

$$H(S_i') = H(S_k) + m \times (i-k) \tag{7}$$

Where m is the height difference of the sampling point, $H(S_k)$ is the elevation of the point S_k. If $H(S_i') > H(S_i)$, the current point S_i is not visible, and then steps to the next point until $H(S_i') < H(S_i)$. With the above algorithm, the backward iteration calculation of the sampling points on line of sight can finally obtain the visual point set $\{S_i | i = 1, 2 \cdot sn\}$ of all sampling points between V and T, which not only ensures the analysis accuracy but also improves the execution efficiency.

Viewshed is the extension of intervisibility, which generates multiple LOS with viewpoint V within the analysis range to cover the visual range. Based on LOS VT, the whole terrain area is scanned by rotating around the elevation axis atα° for many times to obtain the terrain curve set $\{L_i | i = 1, 2 \cdot sn\}$, as shown in Figure 6 (b). $L_i$is sampled equidistance in the horizontal direction, and the visibility of each sampling point is calculated by above algorithm. Finally, the visibility result within the analysis range is obtained and quickly and accurately rendered on the view by Cesium. It should be noted that α and the sampling interval Δl can be selected according to the range and actual situation, and it is generally recommended that Δl be less than the resolution of DEM.

3.3.4 Path Planning Combined with Slope Weighting. In the initial stage of island planning and development, the existing WebGIS could not provide users with path planning due to the lack of public transportation network. The traditional A* algorithm [11] only takes the spatial Euclidean distance as the estimated cost of the algorithm in 3D island with complex terrain. Although the planned path is the shortest, it is not necessarily suitable for people to walk or car to drive. Therefore, this paper proposes an A* path planning algorithm combining slope weighting based on the island topography and DEM data. Without considering the surface vegetation and roughness, it can plan a mountain path with gentler slope (<36°) and shorter distance as possible. Distance and slope factors are added to the original algorithm to optimize the path search. The path search evaluation functions of the improved algorithm are shown in Equation 8). Besides, to improve the search efficiency while ensuring the rationality, $h(n)$ is weighted and efficiency increases as the w_h becomes larger. For this reason, $h(n)$ is given more weight at the beginning of the search, and w_g and w_h are gradually balanced as the search deepens.

$$\begin{cases} f(n) = w_g \times g(n) + w_h \times h(n) \\ g(n) = g(n-1) + [1 - (3.123 \times 10^{-3}) \cdot S_n - (6.948 \times 10^{-4}) \cdot S_n^2)]^{-1} \cdot L \\ h(n) = (1 + \tan^2 S_T) \cdot (1 + \tan^2 S_n) \cdot (dis_T + h_T) \end{cases} \tag{8}$$

Where $f(n)$, $g(n)$, $h(n)$ are the total cost function, actual cost function and estimated cost function respectively. S_n is the slope of the current node n and its parent node $(n-1)$. L is the space straight-line distance between nodes. S_T is the slope between the current node and the target node; dis_T is the horizontal distance between the current node and the target node; h_T is the absolute value of the elevation difference between the current node and the target node. After many tests, the initial w_{h0} and $w_{h\max}$ are set as 0.5 and 0.8 respectively.

In the process of path search, the improved algorithm needs to continuously expand nodes from the current node to the surrounding nodes and add them to the OpenList that is a list of nodes to be detected during the search. To realize the rapid update of nodes in the large island scene, another improvement is made. We used the minimum heap in OpenList to optimize the data structure to quickly filter and eliminate the nodes with the maximum f, so as

Figure 7: System Home Page.

to continuously search forward and finally find the best path that meets the conditions efficiently.

4 SYSTEM FUNCTION REALIZATION AND TESTING

4.1 Development and Test Environment

This system is developed on Windows10 64-bit system, CPU is the Intel Core i5 8^{th} 8G and GPU is the NVIDIA GeForce MX250. Development language mainly uses JavaScript, HTML and CSS, development framework includes Cesium, jQuery and Vue. Visual Studio Code is used as the system development tool, and Google browser for debugging and testing. The DEM data accuracy in the test was 30m which uses CesiumLab tool provided by the official Cesium to perform grading slicing and data publishing of elevation data. Besides, we use Node.js as the local Web server to publish files.

4.2 Main Function Realization Effect

The home page of system is shown in Figure 7, which is divided into four areas. The left side includes the system function selections and the center is used to show 3D earth scene and analysis results. The bottom area can display some regular real-time information such as position, elevation and hierarchy and so on. The right side is dominated by some common scene mode controls. The overall system page is friendly and in line with the user operation habits. Users can select the required functions for island browsing, terrain and spatial analysis. The implementation effect of some functions is shown in Figure 8

4.3 Performance Analysis

After lots of testing, the overall system performance has been improved and performed well, mainly reflected in the following aspects. Firstly, the adaptability of the system is strong. The system can be visualized in different browsers with better scalability and cross-platform characteristics. Additionally, the response speed of the system is fast. From WebGL hardware acceleration to optimizing data structure then to improving some technology algorithms, the system has effectively improved rendering capability and analysis speed comparing to some GIS software. Last but not least, the accuracy of the system has improved. Cesium applies the precise WGS84 ellipsoid model to simulate the earth's surface, and uses its internal

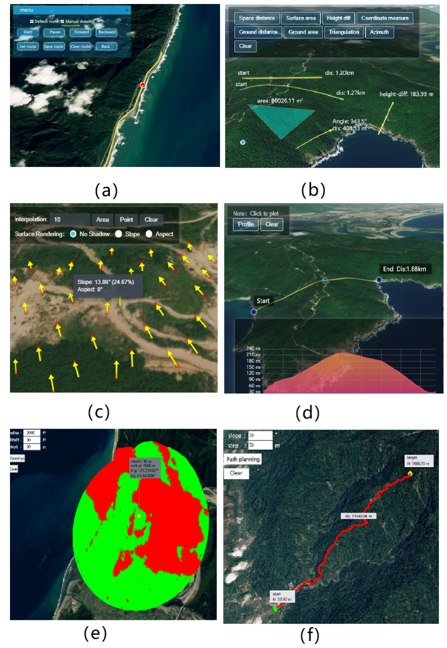

Figure 8: Some Functional Implementation of the System. (a) Live Flight Roaming. (b) Terrain Measurement. (c) Slope and Aspect Analysis. (d) Profile Analysis. (e) Viewshed Analysis. (f) Island Path Planning.

high-precision longitude and latitude coordinates transformation to ensure the accuracy of the island surface position. Besides, through interpolation optimization and elevation correction, the sampling points are closer to the real ground elevation. The optimization algorithm combined with the special island terrain factors, such as visual analysis, can achieve the comparative high-precision analysis effect as R_3 algorithm [10] and SuperMap in a shorter time.

5 CONCLUSION

This paper presents the design of island GIS. It combines the current latest WebGIS technology and Cesium open source framework. The system adopts B/S architecture and modular development idea, takes key technologies as the support to carry out secondary Cesium development, and finally designs a lightweight, cross-platform high-precision WebGIS system for island application. The system fully realizes the functions of 3D map browsing, terrain measuring and plotting, spatial analysis. Through the test, the overall system performance is satisfactory, and it can browse and analyze 3D island scenes efficiently. In the future, we will continue to expand the system functions based on the island features, constantly optimize the system performance and improve user experience.

REFERENCES

[1] Guiseppe Modica, Maurizio Pollino, *et al.* (2016). Land Suitability Evaluation for Agro-forestry: Definition of a Web-Based Multi-Criteria Spatial Decision Support System (MC-SDSS): Preliminary Results. Computational Science and Its Applications, pp. 399-413.

[2] Xiaopeng Sun. (2018). 3D Scene Construction Method Based on Cesium. js and Map World. Geospatial Information, p. 01.
[3] Bing He, Wenxiong Mo, *et al.* (2018). Development of power grid Web3D GIS based on Cesium. 2016 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), pp. 2465-2469.
[4] Murshed S M, Al-Hyari A M, Wendel. (2018). Design and implementation of a 4D web application for analytical visualization of smart city applications. ISPRS International Journal of Geo-Information, p. 276.
[5] Nistor M-M, Rahardjo H, Satyanaga A, *et al.* (2019). GIS-based approach to identify the suitable locations for soil sampling in Singapore. Geographia Technica, p. 14.
[6] Guoqiang Yang,Yang Jin,Yun Li,et al. (2020). GIS-based geological assessment for engineering construction in the Meishan Island, Ningbo. Marine Sciences. pp.133-140.
[7] Junfu Fan, Taoying Hu, Xinhui He, Liu Qin, Guihua Li. (2019). Multi-source digital map tile data mashup scheme design based on Cesium. Journal of Remote Sensing, pp. 695-705.
[8] Changjun Huang, Hongmei Xia, Ziwei Du. (2018). Discussion on Different DEM Interpolation Methods in Typical Hilly Area of Southern China. Journal of Hunan City University (Natural Science Edition), pp. 1-6.
[9] Yang S, Yang N, Chen C F. (2021). Accuracy assessment and improvement of SRTM DEM based on ICESat/GLAS under the consideration of data coregistration over Jiangxi Province. Journal of Geo- information Science, pp. 869- 881.
[10] Hao-Nguyen T, Duy T N, Tra-Duong A. A new algorithm for viewshed computation on raster terrain. (2018). 2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications & Computing (SigTelCom). IEEE, pp. 56-60.
[11] Korkmaz M, Durdu A. (2018). Comparison of optimal path planning algorithms.2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), pp. 255-258.