Public release of the GSI Maps 3D on the Internet

Noriyuki TAKAKUWA, Shoichi OKI, Hidenori FUJIMURA, Ritsu OKAYASU, and Takenori SATO

(Published online: 26 December 2014)

Abstract

The GSI uses contour maps to express topography on planar maps, but users need a certain ability to read and understand the topography. Therefore, the GSI developed the GSI Maps 3D that enables users to view any area in Japan in 3D on an Internet browser and to create the data for 3D printers, so that anyone can intuitively understand topography.

The GSI Maps 3D site acquires elevation tiles delivered from the GSI Maps web server for the surrounding area which the user selects, and create a 3D model using a technology called WebGL. Then, the site displays 3D map by attaching corresponding topographical map or aerial photo images to the ground surface. This 3D processing utilizes the latest Internet technologies such as WebGL and HTML5 Canvas, and various technical issues associated with these technologies have been addressed. Regarding functions that are output as data for 3D printers, the GSI received advice from outside experts on such things as file formats.

Future plans call for such public infrastructure as dams and bridges to be displayed in GSI Maps 3D and to improve functions for managing it.

1. Introduction

The GSI provides maps and aerial photographs through such media as paper, CDs, DVDs, and Internet. However, in nearly all cases these maps and aerial photographs show conditions of the ground surface as seen from directly above, and so users must have a certain ability to recognize actual space and shape, such as topography that is depicted by these maps and aerial photographs. Therefore, maps and aerial photographs shown in 3D have an advantage of enabling an intuitive understanding of such things as ground and topographic conditions.

Since 1993, the GSI has been providing Digital Elevation Models (DEMs), and combining DEMs with maps and aerial photographs to enable the creation of bird's-eye view maps using GIS and other software. However, this work requires a certain level of knowledge and skills for using the software, so it is never an easy task.

At the same time, recent advances in many web browsers have enabled them to utilize 3D computer graphics (3D CG). Furthermore, the use of 3D printers has been growing rapidly in the printing technology world.

Against this backdrop, the GSI has developed the GSI Maps 3D system which can combine the DEMs

Fig.1 An example of GSI Maps 3D (Mt. Tsukuba)

with maps/aerial photos, and easily create and show maps expressed in 3D on an Internet browser. This system was publicly released in March 2014 (Fig. 1).

Even if they don't have special knowledge or software, anyone can use this new system to create 3D maps of freely selected areas in Japan. Maps expressed in 3D makes it easy to understand topography, so there are great expectations for a wide variety of applications for this system.

However, the implementation of these 3D maps will require technologies and techniques that are different from what the GSI has used to create simple flat maps. The following is an explanation about the WebGL technology for 3D expression with browsers, and issues, etc., associated with its use.

2. WebGL

As design guidelines for GSI Maps 3D, there is a method for expressing in browsers the results of 3D calculations made on the server side. However, this would place a great burden on servers, and responsiveness would be greatly reduced. Therefore, it was decided to have all processing done on the user side.

WebGL that is used in the GSI Maps 3D is a technology that does not require special plug-ins for browsers, nor does it need usage fees to support 3D CG processing on an Internet browser. Specifically, it is a technology that connects JavaScript with OpenGL ES 2.0, which supports 3D computation with a specific hardware (Fig. 2). It works with such browsers as Internet Explorer 11, Firefox, Google Chrome and Safari.

The initial stage of development began with technical examinations about the use of WebGL.

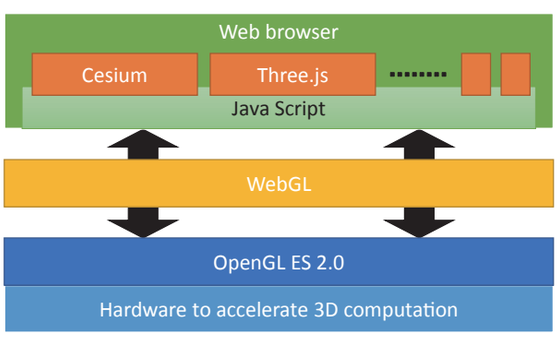

Fig. 2 Browser and layer structure

2.1 Cesium

When using WebGL from JavaScript, it is possible to use various JavaScript APIs to match the purpose.

The first API that was considered was Cesium, which is used in the moon globe released in December 2013 that can be freely rotated (Fig. 3).

Cesium enables polygons and other 3D elements to be designated and shown using latitude and longitude, and has various APIs tailored to geospatial information. Furthermore, there is also sample code that can line up and attach GSI map tiles to a globe as technical

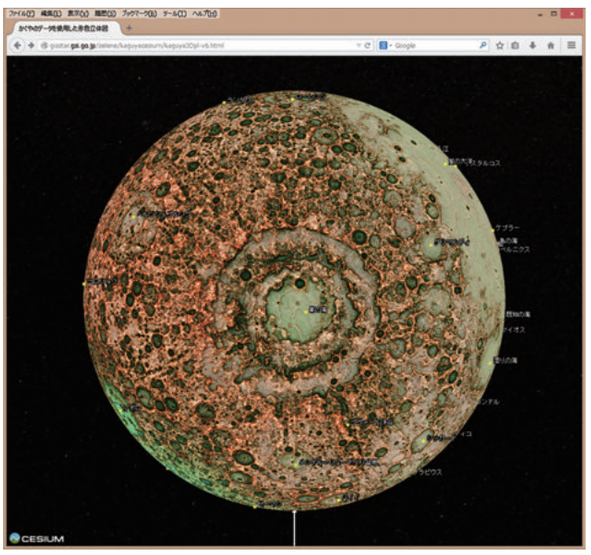

Fig.3 Moon globe using Cesium

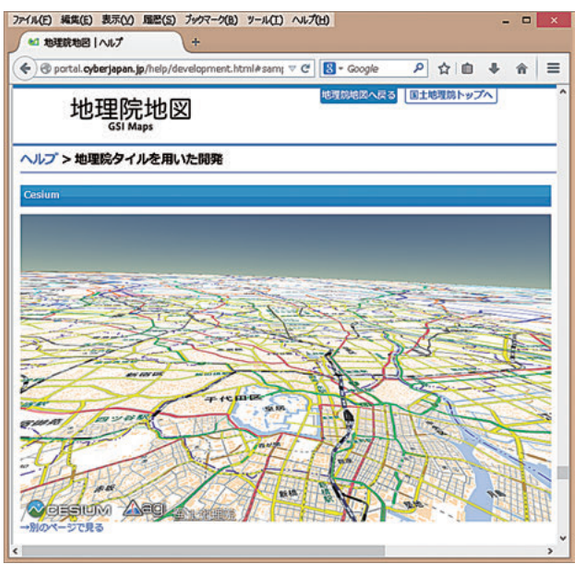

Fig.4 GSI map using Cesium

information of GSI maps (Fig. 4).

However, while Cesium has a high level of APIs, the operating environment can be limited by the functions that are used.

Moreover, there are few documents that can convert elevation tiles for expressing topography into dedicated data for Cesium, and no method has yet been found to satisfactorily resolve this issue.

For such reasons, a search was made for other APIs.

2.2 three.js

Although three.js is an API that may not be as tailored to geospatial information as Cesium is, it can utilize a high level of functions needed for creating models, ray tracing, even animation as 3D CG.

The first use of three.js at the GSI was in the localized 3D maps of the moon and Nishinoshima Island that were released in December 2013 (Fig. 5).

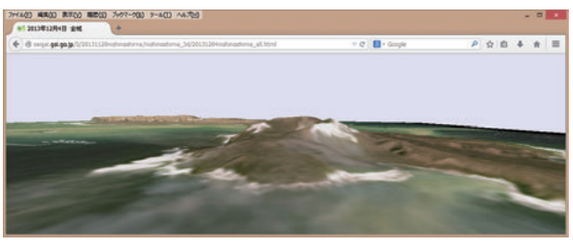

Fig.5 3D map of Nishinoshima Island using three.js

Given the elevation data and limited operating environment, three.js was adopted for the GSI maps 3D.

3. Basic elements and processing flow of 3D CG

The basic elements of 3D CG are object (mesh), light source (lighting), and point of view (camera) (Fig. 6).

Mesh also contains geometry that expresses shapes as polygons, and material that expresses the texture of the surface.

Geometry is not only something that expresses shapes with apexes and planes, but is also something that suppresses shadowing by setting normal vectors toward apexes and planes.

Material can express a texture of the surface by designating reflection and scattering of light and colors,

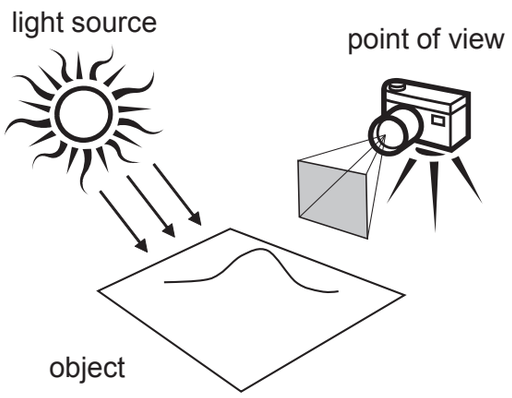

Fig.6 Basic elements of 3D CG

and can also attach image data (texture images) to surfaces.

Based on the information of these elements, calculations simulating behavior of rays called “ray tracing” are made. The results are displayed on the browser screen as 3D CG.

The following shows the processing flow of the JavaScript programming of three.js.

1) Initialization
   The WebGL objects with the scope of display on the web browser, etc., are designated and initialized.
2) Creation of topographic geometry
   For each apex of lattice-like polygons, a Z-value for elevation is entered to create geometry for expressing the shape of the ground surface.
3) Creation of texture image to express the ground surface
   Image of map or aerial photograph is attached to the topographic geometry as materials. While the direction of attachment, projection method, position, etc., can be set in detail, in the case of simple lattice-like geometry, it is possible to simply overlay a single image on the entire topographic geometry (Fig. 7).
4) Creation of mesh data
   Mesh data are created by combining geometry with material.
5) Setting of the lighting
   When there is no light, ray tracing calculations cannot be made, resulting in a simple 3D model that either becomes invisible in the dark or has neither shading nor material feel.
   Light sources can include light that has no directionality

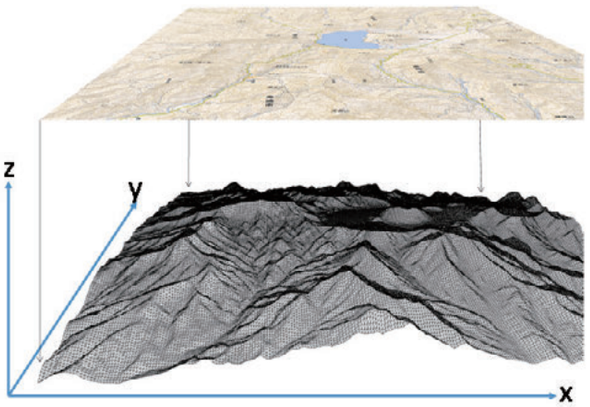

Fig.7 Topographic geometry and texture

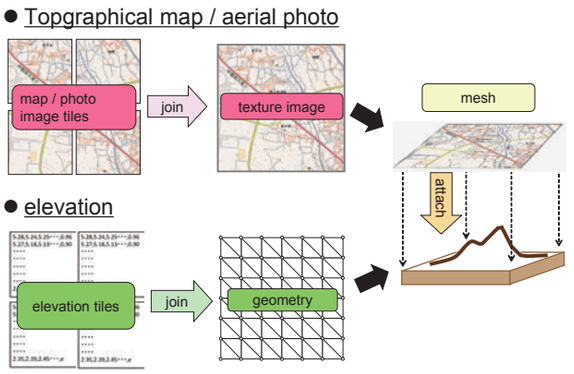

Fig.8 Joining and attachment of GSI tiles

that provides illumination to the entire screen (environmental light), and light with directionality such as parallel light rays and spotlights. The addition of shadows and shade require a light source with directionality.

GSI Maps 3D sets both environmental light sources and parallel light sources.

6) Setting of the camera

The camera, which provides point of view, is set. While objects can be viewed from various directions by rotating and moving the mesh, the same effect can be achieved by fixing the mesh in place and rotating and moving the camera.

7) Rendering

Finally, a 3D model is created based on the set elements.

It should be noted that because the map image texture of the GSI Maps 3D is already shaded, it is set so that ray tracing calculations for the ground surface mesh based on light source information are not made, and that the shadowing of images and shadowing based on calculations are not expressed redundantly.

4. Use of GSI tiles

The topographic geometry of the GSI Maps 3D system creates data from elevation tiles. Also, the texture images that are attached to the ground surface are created from topographical map tiles or aerial photo tiles. These tiles are all delivered from the GSI maps web server on the Internet (Fig. 8).

Regarding the geometry that expresses topography, grid-like polygons of selected area are created by joining elevation tiles in text format (comma-separated values) in which elevation values are lined up in lattices of 256 rows x 256 columns. The fineness of elevation tile for expressing topographical surface changes by zoom level, so a tile at a larger zoom level is finer even if its area is narrower.

The number of tiles that are read into a browser is either 9 (3 x 3) or 16 (4 x 4). Because the processing load of browsers increases, and responsiveness decreases, as the number of geometry apexes increases, the data of the joined elevation tiles is thinned out to ultimately create a latticed geometry of 257 x 257 apexes.

For texture image, image tiles such as topographical map tiles or aerial photo tiles, are used. When they are merged into the same zoom level, elevation tiles only exist from zoom level 0 up to 14, even if detailed map images corresponding to a 1:25,000 scale topography are set for zoom levels 15 to 17. In order for items to be read at a zoom level that is one level higher than the elevation tiles, the number of tiles for a texture image is increased to either 36 (6 x 6) or 64 (8 x 8), so that detailed map at zoom level 15 can be displayed in 3D using elevation tiles at zoom level 14.

4.1 Restrictions on image processing when using Canvas

The joining of texture images uses image-processing functions of JavaScript called Canvas that is

implemented as HTML5 specifications. However, with this Canvas, there is a security restriction for images in cross-domains.

For example, Fig. 9 shows tiles derived from the domain aaa.go.jp being processed on the Canvas of another domain, bbb.go.jp. Images that are processed with this Canvas cannot be forwarded to WebGL or another Canvas.

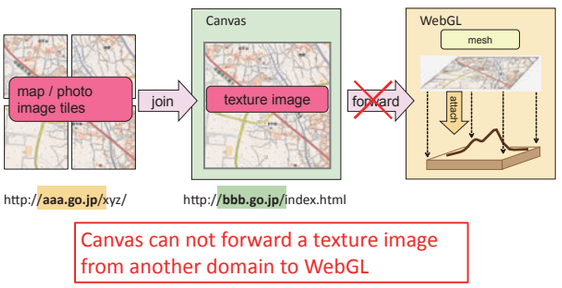

Fig. 9 Cross-domain restriction of Canvas

When such a restriction for a cross-domain exists, it is always necessary to set map and photo tiles on identical servers that have Canvas. Thus, the GIS Maps 3D site is currently built on a hosting server that delivers map / photo tiles.

5. Provision of data for 3D printers

Nowadays, 3D printers that mold directly form 3D digital data have come down in price basically due to the expiration of patents, which has made it more available to individuals. Most of the low-priced 3D printers (Fig. 10) use heat to melt thermoplastics (e.g. PLA, ABS) that are stacked little by little upon each other. Although such Fused Deposition Modeling (FDM) printers can output shapes, they cannot output texture images.

The GSI Maps 3D outputs downloaded files compatible with two types of formats: STL (extension ".stl") and VRML (extension ".wrl") to be printed out with 3D printers, which are formed as the 3D model displayed on a browser.

Fig. 10 Low-cost thermal fusion stacking 3D printers

Files in STL format are for the 3D printers described above that can only output shapes without a texture image (Fig. 11).

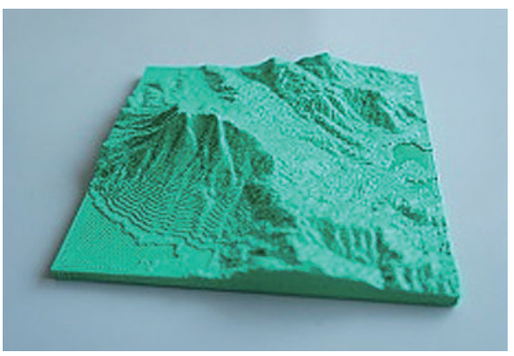

Fig. 11 Mt. Tsukuba output in STL format

Files in VRML format are for 3D printers that can handle color printing, including texture images attached to ground surfaces (Fig. 12), but such printer units are extremely expensive. It appears that in nearly all cases where individuals want to print 3D models in color, they must order printing with 3D printing professionals.

Files for 3D printers cannot be created with

Fig. 12 Nishinoshima Island output in full color in VRML format

JavaScript on a web browser, so a dedicated web server at the GSI creates and sends them to a user whenever a request is received.

Furthermore, as an application of monochromatic 3D models by 3D printers, consideration was given to a method called "projection mapping" which uses a video projector to project maps, aerial photos and overlaid data onto the ground surface of a 3D model as presentations (Fig. 13). One advantage of this system is that even with inexpensive 3D models, various data can be shown and changed on site.

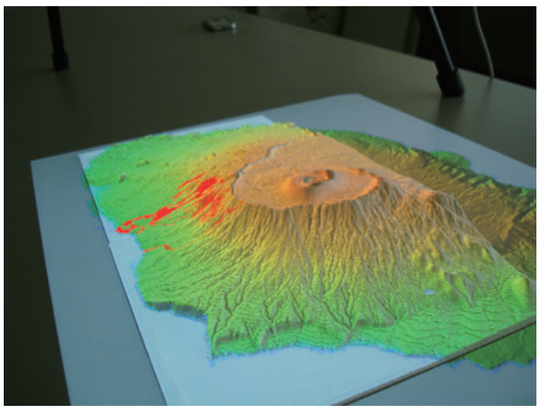

Fig. 13 Projection mapping onto a 3D model
Display of an altitude-tinted elevation map of Izu-Oshima Island overlaid with landslide coverage.

6. Concluding remarks

The public release of GSI Maps 3D is making it possible for anyone to view traditional planar maps in 3D. This enables users to get a more intuitive understanding of topography.

Plans call for a virtual earth globe to be accessible with a web browser which enables users to view topography in 3D with a seamless rendering as if it were real-life, and to add their own data on the surface of the globe.

References

Khronos Group (2013): "WebGL Specification Version 1.0.2.3", https://www.khronos.org/registry/webgl/specs/1.0/ (accessed 20 Jun. 2014).