

SPECIFICATION

Title of the invention: LOD APPLICATION-BASED DATA LIGHTENING METHOD, DEVICE, AND COMPUTER-READABLE RECORDING MEDIUM FOR IMPLEMENTING 3D MODELING ON WEB PLATFORM

TECHNICAL FIELD

[1] The present disclosure relates to a technology for implementing 3D modeling on a web platform, such as visualizing 3D modeling through a web browser supporting HTML5, and more particularly, to a technology for enabling efficient visualization similar to actual 3D modeling data by classifying objects included in 3D modeling data on the basis of LOD and classifying layers thereof and then classifying and applying an object subject to weight lightening, based on each LOD and each layer.

BACKGROUND ART

[2] Digital twin technology allows digitization of various real models, and thus, for example, response to changes in the real environment may be made and the energy flow may be predicted. On the other hand, in a city control monitoring system, facilities and buildings existing in various cities are modeled and used for control.

[3] On these various platforms, visualization is implemented through 3D modeling of buildings, facilities, and inside objects, and output to a user terminal

such as that of a manager to perform various tasks by using digital twins or provide city control services.

[4] However, since it is in most cases necessary to use a dedicated program to visualize a 3D model, a method has recently been developed to visualize a 3D model through a platform that supports HTML5, such as a web browser, so as to enable identification of the model without a specialized program. In addition to these developments, a higher level of visualization is demanded, and the need for visualization technology to meet the demand is increasing, accordingly.

[5] However, in a browser that meets the HTML5 standard, there is a limit to the volume of data that can be processed in a PC environment, so 3D data that can be visualized is also limited. This is because it is difficult to expect the performance of visualizing large-capacity 3D models on a typical PC.

[6] There are several ways to express 3D on the web. For example, in Korean Patent Application No. 10-2013-0138972, a technique for matching CAD data and measurement data to implement the same on the web is proposed. However, in these technologies according to the related art, when visualization is done on a browser that satisfies HTML5 and by using WebGL, there is a limitation on the performance in 3D visualization when 3D data itself is large or heavy. There are browsers that support GPU, but due to the limitations of the web and browsers, it is difficult to expect performance as much as that of typical PC programs.

DETAILED DESCRIPTION OF THE INVENTION

TECHNICAL PROBLEM

[7] The present disclosure provides a technology that enables efficient data weight lightening and thus enables sufficient visualization of 3D models on the web, in particular, in a system requiring monitoring of inside and outside objects in an area, such as city control monitoring, by lightening the weight of modeling data by compressing the modeling data itself as much as possible and enabling to effectively select data that may be weight-lightened.

TECHNICAL SOLUTION

[8] To achieve the purpose described above, an LOD application-based data weight lightening method for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure relates to an LOD application-based data weight lightening method for implementing 3D modeling on a web platform implemented using a computing device including one or more processors and one or more memories for storing instructions that are executable by the processors, the method including: an object classification operation of classifying objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects; an object selection operation of selecting an object to be subject to weight lightening, according to the level and the layer set to each object; and a weight lightening operation of generating weight-lightened second data by compressing modeling data of the object subject to weight lightening, through polygon compression.

[9] The object classification operation may include: an object identification operation of identifying a detailed polygon mesh included in the first data subject to weight lightening, as each object model of the first data, by objectifying the detailed polygon mesh; a layer classification operation of determining, by using a boundary of each object and a reference boundary that is the boundary for a wall object, whether each object is an inside object or an outside object based on the reference boundary, and classifying each object as one of an outside objects layer and an inside objects layer; and a level classification operation of classifying a level of each object, wherein the level of each object is classified based on different standards from each other according to layers set to the objects.

[10] In the object selection operation, by using the level set to each object, when the modeling form of an object is of a level that can be expressed in an atypical form, the object is selected as an object subject to weight lightening.

[11] In the object classification operation, when the size of a boundary of an object classified as an outside objects layer is smaller than the size of the reference boundary, the object may be classified as a lower level than an inside objects layer of the same size.

[12] In the object classification operation, the smaller the size of the boundary of an object, the object may be classified as having a lower level.

[13] In the object classification operation, objects may be classified into levels from 0 to 4 from the top according to the LOD classification of CityGML.

[14] The method may further include, after performing the weight lightening operation, a visualization operation of, upon receiving a request input for visualizing the second data on a web platform, decoding a weight-lightened object

and thus visualizing the model of the object, and outputting the visualized model so that 3D modeling is output on the web platform.

[15] In the visualization operation, when a camera view reaches each level according to a visualization input, the object may be visualized by decoding modeling of a compressed object.

[16] An LOD application-based data weight lightening apparatus for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure, relates to an LOD application-based data weight lightening apparatus for implementing 3D modeling on a web platform implemented using a computing device including one or more processors and one or more memories for storing instructions that are executable by the processors, the apparatus including: an object classification module which classifies objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects; an object selection module which selects an object to be subject to weight lightening, according to the level and the layer set to each object; and a weight lightening module which generates weight-lightened second data by compressing modeling data of an object subject to weight lightening, through polygon compression.

ADVANTAGEOUS EFFECTS OF DISCLOSURE

[17] According to the present disclosure, a method is used, in which objects

constituting a model are identified based on a boundary value, and levels of the sizes of the objects are classified according to the relative sizes thereof, based on a wall of a building, which is the standard for modeling visualization, and whether weight-lightening is possible during visualization is determined based on the levels and the layers used to determine whether an object is an outside or inside object, and an object that may be weight-lightened is weight-lightened by selectively compressing the same.

[18] In this case, except for objects that do not have large capacity because simple modeling thereof is possible, it is highly likely that a modeling form of objects is expressed not in a simple form and thus objects having high capacity efficiency are selectively compressed during compression, and during visualization, while maintaining the form of the model as much as possible, those objects that may be weight-lightened may be efficiently compressed.

[19] Accordingly, the shape of raw data may be maintained as much as possible during visualization, and at the same time, the capacity thereof may be lightened as much as possible, thereby providing 3D modeling data that may be sufficiently visualized on the web.

BRIEF DESCRIPTION OF DRAWINGS

[20] FIGS. 1 to 3 are flowcharts of an LOD application-based data weight lightening method for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure.

[21] FIG. 4 is a structural block diagram of an LOD application-based data

weight lightening apparatus for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure.

[22] FIG. 5 is a diagram for explaining an example in which layers and levels of objects are classified according to an embodiment of the present disclosure.

[23] FIG. 6 illustrates an example of a naming that is set to an object as layers and levels of objects are classified according to an embodiment of the present disclosure.

[24] FIG. 7 is a diagram for explaining a configuration example of first data and second data, according to an embodiment of the present disclosure.

[25] FIG. 8 is an example of the internal configuration of a computing device according to an embodiment of the present disclosure.

BEST MODE

[26] Hereinafter, various embodiments and/or aspects are disclosed with reference to the drawings. In the following description, for purposes of explanation, numerous specific details are set forth to provide a thorough understanding of one or more aspects. However, it will also be recognized by one of ordinary skill in the art that such aspect(s) may be practiced without these specific details. The following description and accompanying drawings set forth in detail certain illustrative aspects of one or more aspects. These aspects are illustrative, however, and some of the various methods in principles of various aspects may be employed, and the descriptions set forth are intended to include all such aspects and their equivalents.

[27] As used herein, “embodiment”, “example”, “aspect”, “exemplary”, etc. are not to be construed as advantageous or advantageous over any aspect or design described herein. It may not be.

[28] Also, the terms "comprises" and/or "comprising" mean that the feature and/or element is present, but the presence or absence of one or more other features, elements and/or groups thereof. It should be understood that additions are not excluded.

[29] Also, terms including an ordinal number, such as first, second, etc., may be used to describe various elements, but the elements are not limited by the terms. The above terms are used only for the purpose of distinguishing one component from another. For example, without departing from the scope of the present disclosure, a first component may be referred to as a second component, and similarly, a second component may also be referred to as a first component. "and/or" includes a combination of a plurality of related listed items or any of a plurality of related listed items.

[30] In addition, in the embodiments of the present disclosure, unless otherwise defined, all terms used herein, including technical or scientific terms, are generally used by those of ordinary skill in the art to which the present disclosure belongs, have the same meaning as understood. Terms such as those defined in a commonly used dictionary should be interpreted as having a meaning consistent with the meaning in the context of the related art, and unless explicitly defined in an embodiment of the present disclosure, these terms are not ideally or excessively construed as having formal meanings.

[31] FIGS. 1 to 3 are flowcharts of an LOD application-based data weight

lightening method for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure. FIG. 4 is a structural block diagram of an LOD application-based data weight lightening apparatus for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure. FIG. 5 is a diagram for explaining an example in which layers and levels of objects are classified according to an embodiment of the present disclosure. FIG. 6 illustrates an example of a naming that is set to an object as layers and levels of objects are classified according to an embodiment of the present disclosure. FIG. 7 is a diagram for explaining a configuration example of first data and second data, according to an embodiment of the present disclosure.

[32] It will be understood that the LOD application-based data weight lightening method for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure, may be performed by a computing device which is to be described later with reference to FIG. 8 and by components of an LOD application-based data weight lightening apparatus for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure, wherein the apparatus may be implemented by a computing device.

[33] First, as illustrated in FIG. 1, in regard to performing the LOD application-based data weight lightening method for implementing 3D modeling on a web platform, according to respective embodiments of the present disclosure, the computing device performs an object classification operation (S10) of classifying objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data

including 3D modeling objects.

[34] In the present disclosure, the LOD refers to a leveling unit for classifying the degree of details in various 3D modeling, that is, for classifying the number of polygons for modeling, for example. For example, in the present disclosure, the level of an object is classified by using LOD which is divided into five stages in City-GML, which is an xml-based data format for storing and exchanging a three-dimensional virtual city model. This is because, a model subject to weight lightening is a very detailed and atypical model, and thus, the lower a level of an object, the number of levels of which increases, the higher the likelihood of the object becoming subject to weight lightening.

[35] In the present disclosure, operation S10 may include a detailed embodiment in which data included in the first data, which is 3D modeling data, is identified as each object model, and naming of and naming of each object model is performed through LOD application-based level designation and layer designation for each identified object model.

[36] That is, as illustrated in FIG. 3, first, the computing device performs an object identification operation (S11) of identifying a detailed polygon mesh included in the first data subject to weight lightening, as each object model of the first data, by objectifying the detailed polygon mesh.

[37] In order to be able to perform operation S11, the first data applied in the present disclosure is to be in a state in which classification by objects is possible; for example, in the case of typical building modeling, modeling may be made by objectifying each detailed polygon mesh of a building. For example, the objects may correspond to a wall, a window, a ceiling, a door, a staircase, a pillar, and

specific objects that may exist in a building, or an outdoor sign, a facility, and the like.

[38] Next, performed is a layer classification operation (S12) of determining, by using a boundary of each object and a reference boundary that is the boundary for a wall object, whether each object is an inside object or an outside object based on the reference boundary, and classifying each object as one of an outside objects layer and an inside objects layer. Next, a level classification operation (S13) of classifying a level of each object is performed, wherein the level of each object is classified based on different standards from each other according to layers set to the objects.

[39] In a detailed example, FIGS. 5 and 6 are referred. A method of applying LOD to 3D modeling may be manually specified by the human eye, but there is a limit to manually performing this method when the number of objects, to which LOD is to be applied, is large. Thus, whether an object includes an LOD is determined using a basic object boundary. And when classification is not possible based on whether it is included within the boundary, the naming rule of objects is included and used to classify. In the case of a building, a reference boundary that separates the interior and the exterior of the building from each other is designated based on a wall (a wall that can express the exterior without differentiating the exterior and interior walls).

[40] The naming of objects is to be understood as an object naming process for identifying whether an object is subject to weight lightening in respect of performing operation S10 described above. The naming rule of objects is to be designated as "wall_" and the entire boundary of all objects included therein

needs to be calculated. This entire boundary is the reference boundary, which determines whether objects in the rest of the models are included or not.

[41] As illustrated in FIGS. 5 (b) and (c), objects 202 and 203 which are either included within or spanning a reference boundary 101 correspond to inside objects when modeling a building. As in FIGS. 5 (a) and (d), objects 201 and 204 that exist outside the reference boundary 101 correspond to outside objects when modeling a building.

[42] On the other hand, the numerical values of the levels used in operation S13 follow the City-GML LOD levels 0 to 4-rule described above. In typical 3D GIS, the levels of LOD classification (scale and distance, tiling method) may involve too many classifications for LOD depth at the time of LOD visualization (levels 0 to 12, 0 to 16 or more). Thus, although the classification of levels 0 to 4 of CityGML is ambiguous, it is desirable to apply the LOD level classification of CityGML in order to use only the required LOD such that the LOD levels are not divided by an even more complex depth. In this case, the numerical values of 0 to 4 corresponding to respective LOD level classification may be defined as a highest level (0) and a lowest level (4) in order.

[43] In this case, the level classification may be defined as follows.

[44] A. Level 0: A stage where only the appearance of a building appears in the terrain.

[45] B. Level 1: A stage where the outer and inner walls of a building are distinguished; in the expression of building texture, CityGML, a modeling geometry distinguishes a boundary geometry for the exterior in a stage where the exterior is distinguished, but it is unnecessary to additionally distinguish the

boundary geometry in this method for weight lightening.

[46] C. Level 2: A stage where walls and components of the walls (windows, doors, etc.) of a model or a building are classified.

[47] D. Level 3: A stage where object in a model or inside a building are classified, but detailed expression is not required. In some cases, this stage explains the details of the detailed expression. However, in the present LOD classification method for weight lightening, this stage is specified merely as a state where expression is not made, and detailed and non-detailed expression are not differentiated here.

[48] E. Level 4: A stage where there is no distinction between the interior/exterior of a model or a building and objects.

[49]

[50] In other words, considering the numerical values of the levels described above, the higher the level, the simpler modeling is possible, and thus the capacity is not large, and can be excluded from being subject to weight lightening; the lower the level, the higher the possibility of being detailed and atypical modeling, and thus also highly likely to be subject to weight lightening.

[51] That is, in the present disclosure, among the objects classified in operation S13, by using the levels that are set to each object, when the modeling form of an object is of a level that can be expressed in an atypical form, the object may be preferably selected as an object subject to weight lightening in the object selection operation (S20) to be described later.

[52] When the size of a boundary of an object classified as an outside objects layer in operation S12 of the present disclosure is smaller than the size of the

reference boundary, the object may preferably be classified as a lower level than an inside objects layer of the same size.

[53] That is, for example, an object classified as an outside objects layer may be realized such that levels from 0 to 4 are all selectable, and this is because a size of an outside object is not designated, for example, the object may be an object with a small size.

[54] In addition, as described above, in the object classification operation in operation S13, in regard to objects that are highly likely to be subject to atypical modeling and require detailed modeling, the smaller the size of the boundary of an object, the object is preferably classified as a lower level.

[55] An example of leveling and layer classification done according to all these object classification operations performed is illustrated in FIG. 6. Referring to FIG. 6, a layer is classified into an outside objects layer and an inside objects layer, and an LOD, that is, a level, is classified according to the boundary size of each object.

[56] In the example of FIG. 6, each level and layer may be classified for each object as follows.

[57] 1. Building Component Visualization Operation - walls, pillars, ceilings, stairs, windows, doors

[58] A. Walls (walls including the exterior and the interior): Levels 0 to 4

[59] B. Pillars, Stairs: Levels 1 to 4

[60] C. Ceilings, stairs, windows, doors: Levels 2 to 4

[61] D. Non-building components - furniture, lights, sensors, etc.: levels 3 to

[62]

[63] 2. For outside objects - roofs, signs, facilities

[64] A. In the case of facilities, when the boundary size does not exceed the reference boundary as in the object 201 in FIG. 5(a), LOD classification may be adjusted to levels 3 to 4.

[65] B. The facilities need to be expressed regardless of the indoor LOD. However, when the object size is too small compared to the reference boundary in levels 0 to 2, the object may not be visually expressed, and thus may be designated as levels 3 to 4.

[66]

[67] When the level and layer of an object are classified through these operations S10, S11 to S13, the computing device performs an object selection operation (S20) of selecting an object to be subject to weight lightening, according to the level and layer set to each object as described above.

[68] Operation S20 of designating an object to be subject to weight lightening is efficient, specifically, when the level is a lower level, that is, mainly when the shape of an object is atypical, as described above. In the above-described example, as objects having a level of 2 to 4 have a modeling form that is not expressed in a simple form in many cases, the objects may be included in objects subject to weight lightening.

[69] Considering this, objects corresponding to an outside layer are highly likely to be classified as a lower level when the objects are of the same size, and thus are likely to be subject to weight lightening.

[70] When objects subject to compression, that is, weight lightening, are

selected based on these factors, the computing device performs a weight lightening operation (S30) of generating weight-lightened second data by compressing modeling data of an object subject to weight lightening, through polygon compression.

[71] In a detailed example, there are several optimization methods for compressing modeling, but 3D modeling compression may be performed using Google's open-source library, Droco, which has a high mesh compression ratio. In the example of the model, the compression efficiency of the LOD levels 2-4 may vary depending on the modeling method, with respect to objects inside a building and facility objects, but as a result of experiments, it was tested that an efficiency of about 80 % to 90 % is possible.

[72] When this weight lightening operation (S30) is performed, as illustrated in FIG. 7, among objects 11, 12, 13 of first data 1, the objects 12 and 13, the level of which is set to a lower level or which are selected as objects subject to weight lightening based thereon, may be realized, in second data 1-1, in a state changed to objects 12-1, 13-1, the model of which is weight-lightened.

[73] After the weight lightening, the 3D model may be visualized by applying the second data in a web browser such as HTML5 described above. That is, as illustrated in FIG. 2, after performing operation S30, when the computing device receives a request input for visualizing the second data on a web platform, the computing device may further perform a visualization operation (S40) of decoding the weight-lightened object and thus visualizing the model of the object, and outputting the visualized model so that the 3D modeling is output on the web platform.

[74] That is, it will be understood that it is assumed that the second data, and not the first data, is already used when performing operation S40. As objects subject to weight lightening are E-encoded through compression, etc., the objects are preferably processed by a decoding process during visualization.

[75] The decoding time should be understood as visualizing the object by performing decoding on the modeling of the compressed object when a detailed view of a camera for visualization reaches each level. That is, the visualization is performed as if an image captured by a camera is output. Here, according to viewpoint processing such as the zoom of a camera, when each viewpoint reaches a level (LOD), a weight-lightened object corresponding to a corresponding level is decoded.

[76] As described above, except for objects that do not have large capacity because simple modeling thereof is possible, it is highly likely that, due to the high possibility that the modeling form will not be expressed in a simple form, objects having a high capacity efficiency are selectively compressed during compression, and objects that may be weight-lightened may be efficiently compressed while maintaining the form of the model during visualization.

[77] FIG. 4 is a structural block diagram of an LOD application-based data weight lightening apparatus 10 for implementing 3D modeling on a web platform, according to an embodiment of the present disclosure. In the following description, descriptions overlapping with those with reference to FIGS. 1 to 3 and FIGS. 5 to 7 will be omitted.

[78] Referring to FIG. 4, the LOD application-based data weight lightening apparatus 10 for implementing 3D modeling on a web platform, according to an

embodiment of the present disclosure, may include an object classification module 11, an object selection module 12, and a weight lightening module 13, and may additionally include a visualization module 14. In the present disclosure, including the first and second data 1 and 1-1, all data and programs used to perform the functions of the present disclosure, such as a weight-lightening model, a level classification model, etc. may be stored in a database 20 and managed.

[79] The object classification module 11 performs a function of classifying objects included in the first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects. That is, the object classification module 11 may be understood as a configuration that performs all functions mentioned in the description of operations S10, S11, S12, and S13 among the descriptions of the embodiments of FIGS. 1 and 3.

[80] The object selection module 12 is a configuration that performs a function of selecting an object to be subject to weight lightening, according to the level and the layer set to each object, and is understood as a configuration that performs all functions mentioned in the description of operation S20 in the description of FIG. 1.

[81] The weight lightening module 13 is a configuration that performs a function of generating weight-lightened second data by compressing modeling data of an object subject to weight lightening, through polygon compression, and is understood as a configuration that performs all functions described with reference to operation S30 in the description of FIG. 1.

[82] The visualization module 14 is a configuration that performs a function of, upon receiving a request input for visualizing the second data on a web platform, through decoding a weight-lightened object and thus visualizing the model of the object, and outputting the visualized model so that 3D modeling is output on the web platform, and is a configuration that performs all functions described with reference to operation S40 in the description with reference to FIG. 2.

[83] FIG. 8 illustrates an example of the internal configuration of a computing device according to an embodiment of the present disclosure, and in the following description, repeated and unnecessary descriptions of the embodiment with regard to the description of FIGS. 1 to 7 provided above are omitted.

[84] As illustrated in FIG. 8, a computing device 10000 may at least include at least one processor 11100, a memory 11200, a peripheral interface 11300, an input/output (I/O) subsystem 11400, a power circuit 11500, and a communication circuit 11600. The computing device 10000 may correspond to a user terminal A connected to a tactile interface device or the computing device B described above.

[85] The memory 11200 may include, for example, a high-speed random access memory, a magnetic disk, an SRAM, a DRAM, a ROM, a flash memory, or a non-volatile memory. The memory 11200 may include a software module, an instruction set, or other various data, which are required for the operation of the computing device 10000.

[86] Here, access to the memory 11200 by other components such as the processor 11100 or the peripheral interface 11300 may be controlled by the

processor 11100.

[87] The peripheral interface 11300 may couple input and/or output peripherals of the computing device 10000 to the processor 11100 and the memory 11200. The processor 11100 may execute the software module or the instruction set stored in the memory 11200 to perform various functions for the computing device 10000 and process data.

[88] The input/output subsystem 11400 may couple various input/output peripherals to the peripheral interface 11300. For example, the input/output subsystem 11400 may include a controller for coupling, to the peripheral interface 11300, peripheral devices such as a monitor, keyboard, a mouse, a printer, or a touch screen or a sensor as needed. According to another aspect, input/output peripherals may be coupled to the peripheral interface 11300 without going through the input/output subsystem 11400.

[89] The power circuit 11500 may supply power to all or some of the components of the terminal. For example, the power circuit 11500 may include a power management system, one or more power sources such as batteries or alternating current (AC), a charging system, a power failure detection circuit, a power converter or inverter, a power status indicator, or any other components for power generation, management, and distribution.

[90] The communication circuit 11600 may enable communication with another computing device by using at least one external port.

[91] Alternatively, as described above, if necessary, the communication circuit 11600 may include an RF circuit to transmit and receive an RF signal, also known as an electromagnetic signal, to enable communication with other

computing devices.

[92] The embodiment of FIG. 8 is only an example of the computing device 10000, and some components illustrated in FIG. 8 may be omitted from the computing device 10000 the computing device 10000 may further include additional components not illustrated in FIG. 8, or may have a configuration or arrangement in which two or more components are combined. For example, a computing device for a communication terminal in a mobile environment may further include a touch screen or a sensor, in addition to the components illustrated in FIG. 8, and may include, in the communication circuit 11600, a circuit for RF communication of various communication methods (WiFi, 3G, LTE, Bluetooth, NFC, Zigbee, etc.). Components that may be included in the computing device 10000 may be implemented as hardware, software, or a combination of both hardware and software, which include an integrated circuit specialized for one or more signal processing or applications.

[93] The methods according to embodiments of the present disclosure may be implemented in the form of program instructions that can be executed through various computing devices and recorded in a computer-readable medium. In particular, a program according to the present embodiment may be configured as a PC-based program or an application dedicated to a mobile terminal. An application to which the present disclosure is applied may be installed in a user terminal through a file provided by a file distribution system. For example, the file distribution system may include a file transmission unit (not shown) that transmits the file according to a request from the user terminal.

[94] The apparatus described above may be implemented as a hardware

component, a software component, and/or a combination of a hardware component and a software component. For example, the apparatus and the components described in the embodiments may be implemented using one or more general-purpose or special purpose computers, like, for example, a processor, a controller, an arithmetic logic unit (ALU), a digital signal processor, a microcomputer, a field programmable gate array (FPGA), a programmable logic unit (PLU), a microprocessor, or any other apparatus capable of executing and responding to instructions. A processing device may execute an operating system (OS) and one or more software applications running on the operating system. The processing device may also access, store, manipulate, process, and generate data in response to execution of software. For convenience of understanding, although one processing device is described as being used, it will be obvious to one of ordinary skill in the art that the processing device includes a plurality of processing elements and/or a plurality of types of processing elements. For example, the processing device may include a plurality of processors or one processor and one controller. Other processing configurations are also possible, such as parallel processors.

[95] Software may include a computer program, code, instructions, or a combination of one or more of these, and may configure a processing device to operate as designed or may independently or collectively give an instruction to a processing device. Software and/or data may be permanently or temporarily embodied in any kind of machine, component, physical device, virtual equipment, computer storage medium or device, to be interpreted by a processing device or to provide an instruction or data to the processing device. The software may be

distributed over networked computing devices, and stored or executed in a distributed manner. Software and data may be stored in one or more computer-readable recording media.

[96] The methods according to the embodiments may be implemented in the form of program instructions that can be executed through various computer means and recorded in a computer-readable medium. The computer-readable medium may include program instructions, data files, data structures, etc. alone or in combination. The program instructions recorded on the medium may be specially designed and configured for the embodiment, or may be known and available to those skilled in the art of computer software. Examples of the computer-readable recording medium include magnetic media such as hard disks, floppy disks, and magnetic tapes, optical media such as CD-ROMs and DVDs, magneto-optical media such as floptical disks, and hardware devices specially configured to store and execute program instructions, such as ROM, RAM, flash memory, and the like. Examples of program instructions include not only machine language codes such as those generated by a compiler, but also high-level language codes that can be executed by a computer by using an interpreter or the like. The hardware devices described above may be configured to operate as one or more software modules to perform the operations of the embodiments, and vice versa.

[97] As described above, although the embodiments have been described with reference to the limited embodiments and drawings, various modifications and variations may be made from the above description by those skilled in the art. For example, an appropriate result may be attained even when the above-

described techniques are performed in a different order from the above-described method, and/or components, such as the above-described system, structure, device, and circuit, are coupled or combined in a different form from the above-described methods or substituted for or replaced by other components or equivalents thereof. Therefore, other implementations, other embodiments, and equivalents to the claims are also within the scope of the following claims.

CLAIMS

1. An LOD application-based data weight lightening method for implementing 3D modeling on a web platform implemented using a computing device including one or more processors and one or more memories for storing instructions that are executable by the processors, the method comprising:

an object classification operation of classifying objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects;

an object selection operation of selecting an object to be subject to weight lightening, according to the level and the layer set to each object; and

a weight lightening operation of generating weight-lightened second data by compressing modeling data of the object subject to weight lightening, through polygon compression.

2. The method of claim 1, wherein the object classification operation comprises:

an object identification operation of identifying a detailed polygon mesh included in the first data subject to weight lightening, as each object model of the first data, by objectifying the detailed polygon mesh;

a layer classification operation of determining, by using a boundary of each object and a reference boundary that is the boundary for a wall object, whether

each object is an inside object or an outside object based on the reference boundary, and classifying each object as one of an outside objects layer and an inside objects layer; and

a level classification operation of classifying a level of each object, wherein the level of each object is classified based on different standards from each other according to layers set to the objects.

3. The method of claim 2, wherein, in the object selection operation, by using the level set to each object, when the modeling form of an object is of a level that may be expressed in an atypical form, the object is selected as an object subject to weight lightening.

4. The method of claim 2, wherein, in the object classification operation, when the size of a boundary of an object classified as an outside objects layer is smaller than the size of the reference boundary, the object is classified as a lower level than an inside objects layer of the same size.

5. The method of claim 2, wherein, in the object classification operation, the smaller the size of the boundary of an object, the object is classified as having a lower level.

6. The method of claim 1, wherein, in the object classification operation, objects are classified into levels from 0 to 4 from the top according to the LOD classification of CityGML.

7. The method of claim 1, further comprising, after performing the weight lightening operation, a visualization operation of, upon receiving a request input for visualizing the second data on a web platform, decoding a weight-lightened object and thus visualizing the model of the object, and outputting the visualized model so that 3D modeling is output on the web platform.

8. The method of claim 7, wherein, in the visualization operation, when a camera view reaches each level according to a visualization input, the object is visualized by decoding modeling of a compressed object.

9. An LOD application-based data weight lightening apparatus for implementing 3D modeling on a web platform implemented using a computing device including one or more processors and one or more memories for storing instructions that are executable by the processors, the apparatus comprising:

an object classification module which classifies objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects;

an object selection module which selects an object to be subject to weight lightening, according to the level and the layer set to each object; and

a weight lightening module which generates weight-lightened second data by compressing modeling data of an object subject to weight lightening, through

polygon compression.

10. A computer-readable recording medium, wherein the computer-readable recording medium has stored therein instructions that cause a computing device to perform the following operations comprising:

an object classification operation of classifying objects included in first data, as one of a plurality of levels and one of outside and inside objects layers, by designating a level of each object through a preset level of detail (LOD) classification algorithm for each object included in the first data including 3D modeling objects;

an object selection operation of selecting an object to be subject to weight lightening, according to the level and the layer set to each object; and

a weight lightening operation of generating weight-lightened second data by compressing modeling data of an object subject to weight lightening, through polygon compression.