J Supercomput (2019) 75:4505–4518
https://doi.org/10.1007/s11227-017-1975-0

A web-based 3D ontology navigation system for spinal disease diagnosis

Ji-Seong Jeong[1] · Min-Ho Song[1] · Sang-Ho Lee[2] · Mihye Kim[3] · Nakhoon Baek[4] · Kwan-Hee Yoo[1,5]

Published online: 11 February 2017

Abstract This paper presents a web-based three-dimensional (3D) navigation system for a spinal disease ontology using a 3D virtual human spine. The 3D navigation system consists of three main modules: a query module for finding spinal diseases and their causes and treatments from the spinal disease ontology via a web page, a 3D rendering module for rendering the virtual human spine on a web browser, and a visualization

✉ Kwan-Hee Yoo
khyoo@chungbuk.ac.kr

Ji-Seong Jeong
songminho@chungbuk.ac.kr

Min-Ho Song
farland83@chungbuk.ac.kr

Sang-Ho Lee
shlee@kisti.re.kr

Mihye Kim
mihye@cu.ac.kr

Nakhoon Baek
oceancru@gmail.com

1 Department of Digital Informatics and Convergence, Chungbuk National University, 1 Chungdaero, Seowongu, Cheongju, Chungbuk 28644, South Korea

2 Software Research Center, Korea Institute of Science and Technology Information, 245 Daehakro, Yuseonggu, Daejeon 34141, South Korea

3 Department of Computer Science and Engineering, Catholic University of Daegun, 13-13 Hayangro, Kyungpook 38430, South Korea

4 School of Computer Science and Engineering, Kyungpook National University, 80 Daehakro, Bukgu, Daegu 41566, South Korea

5 Department of Computer Science, Chungbuk National University, 1 Chungdaero, Seowongu, Cheongju, Chungbuk 28644, South Korea

module to view the retrieved ontology information by connecting it with the virtual human spine. It was implemented using Virtuosos SPARQL, Java, JavaScript, Jena API, JDBC, AJAX, HTML5, WebGL, and SVG using a Virtuoso server on the Web via JSP web pages. Spine specialists can navigate and simulate spinal diseases using the 3D navigation system via virtual experiments based on the ontological interrelations, allowing them to make more accurate diagnoses of spinal diseases.

Keywords Ontology · Spinal disease ontology · Ontology navigation · Ontology visualization · Web-based 3D ontology navigation system

1 Introduction

Recently, a spinal disease ontology system has been developed by the Korean Institute of Science and Technology Information (KISTI), in cooperation with the Catholic University of Korea [1,2]. A virtual human spine was also constructed to allow spine specialists to investigate spinal diseases and treatments via virtual experiments, based on enhanced reasoning capabilities [3–5]. This virtual spine system also has applications in the education of medical students, physicians, and biomedical engineers [6]. The study focused on the development of an ontology for spinal diseases that occur frequently in specific areas of the human spine [7] and contains the anatomy of the spine, causes of spinal diseases, treatments, and classification information related to the spine. It was constructed using Web Ontology Language (OWL) Full by the World Wide Web Consortium (W3C) [8], and using both SemanticWorks by Altova [9] and Protege by Stanford University [10].

Neches et al. defined an ontology as basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary [1,11]. Gruber defined the ontology as a formal, explicit specification of a shared conceptualization for a specific domain [12] where the term conceptualization was defined as a process to identify a set of objects, concepts or other entities in a given domain and the conceptual relationships among them [13]. A formal ontology can consist of concepts and definitions, as well as formal axioms describing relations between them. The definitions associate the components of entities, such as classes (subclasses and attributes), relations, functions and constraints [12,14]. Ontologies support accurate inferences and advanced reasoning services via inferring implied knowledge automatically from the ontological relationships [6]. Additionally, Lee et al. proposed a mobile Web navigation, which could be used to receive any information seamlessly on a mobile device over digital ecosystem [15].

In this paper, we present a web-based system to navigate spinal diseases and their causes and treatments via a 3D virtual human spine using the spinal disease ontology to allow users to make more accurate diagnoses of vertebral column diseases. Spine specialists can easily make various query operations on the spinal disease ontology and can view the retrieved ontology information by connecting it with the 3D virtual human spine. Through our study, we obtained effective spinal disease specimens and images for degenerative grade determination by the degenerative changes of intervertebral disk. We also produced spine images of normal elderly people, including the Xray, Computer Tomography (CT), Magnetic Resonance (MR), and Bone Mineral Density (BMD) productions of neck spine and waist vertebrae. The 3D navigation system for spinal diseases was implemented using SPARQL Protocol and RDF Query Language (SPARQL), Asynchronous JavaScript and XML (AJAX), HyperText Markup Language 5 (HTML5), Scalable Vector Graphics (SVG), and Web Graphics Library (WebGL) in Java Server Pages (JSP) on a Virtuoso server [17–22]. The system consists of three main modules: query module for finding the spinal ontology data via a web page, a 3D rendering module for rendering the virtual human spine via a web browser using WebGL, and an interactive visualization module to view the ontology information retrieved in response to a query by connecting it with the virtual spine. The content of this paper was partially and shortly introduced in previous work [16].

The remainder of the paper is organized as follows. In Sect. 2, we discuss the spinal disease ontology constructed by the Catholic University of Korea [1,2]. In Sect. 3, we describe the web-based 3D navigation system for spinal disease diagnosis based on the ontology. In Sect. 4, we conclude the paper with a discussion and identify directions for future work.

2 Spinal disease ontology

The spinal disease ontology has been developed by the KISTI for vertebral column diseases that occur in specific areas of the human spine to support with treatment and rehabilitation of age-related spinal disorders. To establish the spinal ontology, 20 spinal diseases were analyzed by experts and clinicians in diverse fields including medical informatics, computer science and imaging [1,2]. The anatomical information and disease-related information were represented in the ontology, as shown in Table 1. The anatomical information was classified into two categories, and the disease-related information was classified into five categories. These were extracted from medical publications, and the anatomical definitions were referenced from Stedmans Medical Dictionary [23].

The spinal disease ontology was represented using the OWL full sublanguage and was constructed using SemanticWorks and Protege. The anatomical information consists of 50 classes, i.e., the vertebral column, the 33 vertebrae, and the other 11 components of the spine, where each class has 6 properties. The disease-related information was composed of 21 classes, i.e., 20 classes for each selected disease and one disease class as a superclass of the 20 classes for the 20 selected diseases, where each class consists of 18 properties. Figure 1 shows an OWL ontology graph of the 21 diseases presented in Protege [2].

3 Ontology 3D navigation system

3.1 System configuration

Figure 2 shows an overview of the web-based 3D navigation system for the spinal disease ontology. It was implemented using Virtuoso, quad store and jena for controlling ontology data on a Virtuoso server, using AJAX, HTML5, javascript and JavaBeans for Web Page interaction, and using WebGL [21] and ThreeJS [22] for rendering 3D

Table 1 Information represented in the spinal disease ontology [1,2]

Information	Category	Ontology	OWL Expression
Anatomical information	Location	Anatomical location	spine: isPartOf
	Properties	Part name	rdfs: label
		Anatomical classification	rdfs: subClassOf
		Standard code for the structure	spine: KOSTOM rdfs: label
		Definition of the structure	spine: definition
		Description of the structure	spine: description
Disease related information	Location	Anatomical disease location	spine: hasSite
	Disease properties	Name of disease	rdfs: label
		Classification of disease	rdfs: subClassOf
		Apply the standard code	spine: KOSTOM
		Definition of disease	spine: definition
		Description of disease	spine: description
		Clinical diagnosis	spine: diagnosis
		Cause	spine: hasCause
		Concomitant diseases	spine: hasConcomitantDisease
		Complication	spine: hasComplication
	Symptom/sign	Symptom	spine: hasSymptom
			spine: causeOfSymptom
		Sign	spine: hasSign
	Treatment	Surgical treatment	spine: hasSurgicalTreatment
		Nonsurgical treatment	spine: hasNonSurgicalTreatment
		Conservative treatment	spine: hasTreatmentConservative
	Image	Preoperative image	spine: hasImageBeforeTx
		Postoperative image	spine: hasImageAfterTx

virtual human spine. The navigation system shows information on the human spine and on diseases that can occur in the individual vertebrae, as well as treatments for these diseases in dialog boxes by connecting it with a 3D virtual human spine.

The 3D navigation system was composed of three main modules: a query module for finding the spinal ontology data via a web page, a 3D rendering module for rendering the virtual human spine via a web browser using WebGL, and an interactive visualization module to view the ontology information retrieved in response to a query by connecting it with the virtual spine. Table 2 lists the main features of these three modules.

3.2 Query module

To query and navigate the spinal ontology data using a JavaServer Pages (JSP) web page, we uploaded the OWL files of the spinal disease ontology in Protege to a Virtuoso server and queried the data on the server using a semantic SPARQL query language

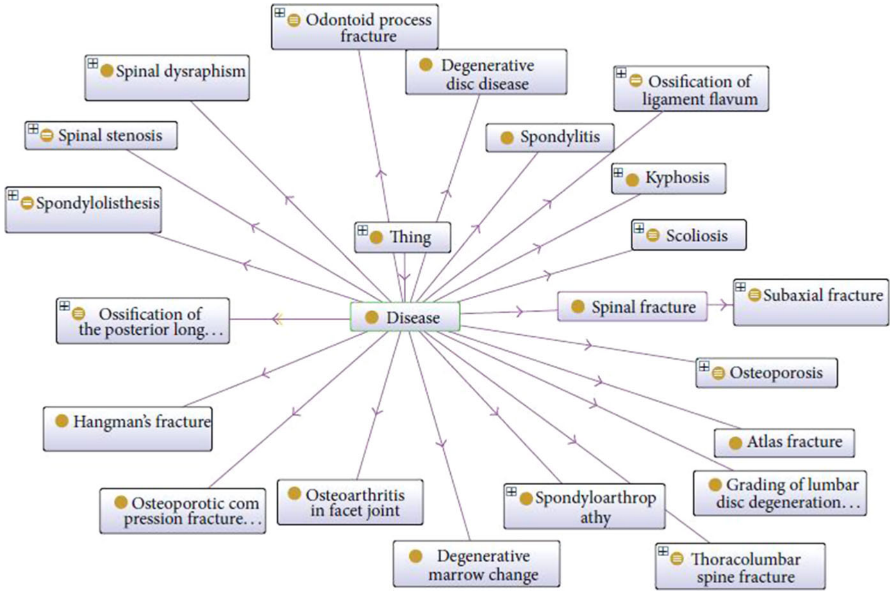

Fig. 1 An OWL ontology graph of the 21 spinal diseases in Protege [2, p. 4]

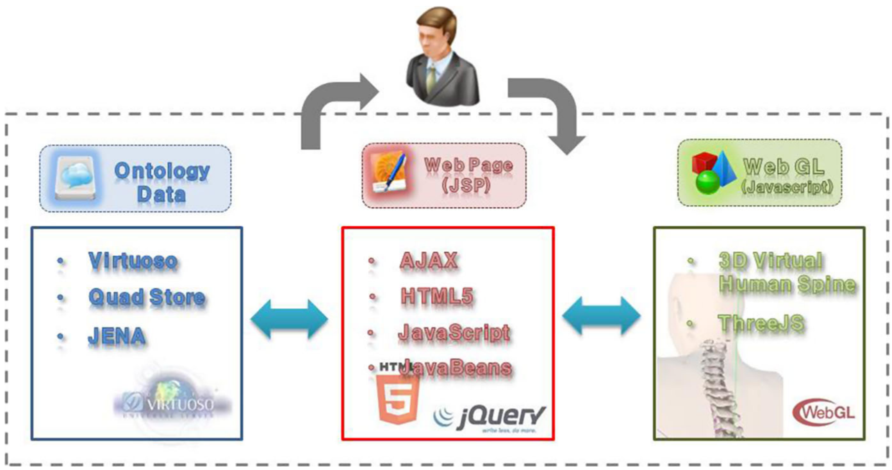

Fig. 2 An overview of the web-based 3D ontology navigation system [16, p. 278]

for the spinal disease databases to retrieve and manipulate data stored in Resource Description Framework (RDF) format.

Virtuoso is a cross-platform hybrid data server that combines Standard Query Language (SQL), Resource Description Framework (RDF), XML, and free-text data management with the functionality of a Web application server in a single system [18,19], and was developed by OpenLink Software under open source license. We installed a Virtuoso server and uploaded the spinal disease ontology (i.e., OWL files)

Table 2 The main features of the three main modules of the navigation system

Module	Main features
Query module	Store the spinal disease ontology in the Virtuoso server
	Query the spinal ontology data using Virtuosos SPARQL
	Implement an ontology query module using Java
3D rendering module	Visualize query results by connecting them with the 3D virtual spine using WebGL
	Move/rotate the position of a camera using a mouse or a keyboard
	Highlight a region of the virtual spine using a mouse
	Set the transparency of the virtual spine through user interactions
Visualization module	Create a dialog box in response to a disease search or a region search
	Generate dialog content (i.e., spinal diseases and their regions and treatments retrieved as a search result) dynamically using AJAX
	Visualize the connectivity between the dialog box and the virtual spine model using SVG
	Visualize the KOSTOM and FMA information

in Protege into the server as Linked Data using the quad store upload feature of Virtuoso. That is, the OWL files are stored in the form of Linked Data to deploy on the Web via the ontology query language (i.e., SPARQL). We set the graph Internationalized Resource Identifiers (IRIs) of a named graph to http local server site. We then obtained the ontology information via a JSP web page using SPARQL and analyzed the spinal diseases, components of the spine, specific spinal diseases that occur in a certain regions of the spine, and information of the parent–child nodes for a specific node in the ontology.

To do it, we provide four kinds of query to retrieve the ontology data: Q1-Get full disease, Q2-Get the full list of spine bone, Q3-Access list of diseases that occur at a specific site, Q4-Access the child and parent information of a node. When we assume that ontology spine disease information is constructed at localhost:8890 and a 3D spine model is stored at localhost:8891, Table 3 shows commands to perform four main queries, Q1, Q2, Q3 nad Q4.

To process the above SPARQL queries and display the query results on the Web, we implemented an ontology query module using Java. Figure 3 shows the overall process of this query module. Users can query and navigate the spinal ontology data via the JSP web page. A query result is displayed in a Web-based dialog box using AJAX and HTML5. We accessed the ontology data, which were uploaded to the Virtuoso server from Protege, using Java Database Connectivity (JDBC). The ontology data consisted of triples comprising a subject, a predicate and an object. These three elements were identified using Jena API. We implemented SPARQL queries using JavaBeans.

3.3 3D rendering module

A 3D virtual human spine was rendered on the Web using WebGL to visualize query results on a virtual human spine. WebGL is a free Web standard 3D rendering API

Table 3 Statements for performing four main queries, Q1, Q2, Q3 and Q4

Query	Statement
Q1	select distinct ?Skorean ?Senglish from <http://localhost:8890/spine_new> where { ?s rdfs:label ?Skorean. ?s rdfs:label ?Senglish. ?s ?p ?o FILTER ((lang(?Skorean) = ’kr’) && (lang(?Senglish) = ’en’) && regex(?o, ’http://61.78.109.24.spine#INSTANCE_’))}
Q2	select distinct ?o ?part from <http://localhost:8890/spine_new> where { ?o rdfs:label ?part. ?s ?p ?o FILTER ((regex(?o, ’INSTANCE’)) && (lang(?part) = ’krpt’))}
Q3	select ?Skorean from <http://localhost:8890/spine_new> where { ?s rdfs:label ?Skorean. ?s ?p ?o FILTER ((regex(?o, ’PartName$’)) && (lang(?Skorean) = ’kr’))}
Q4	select distinct ?s ?p ?o from <http://localhost:8890/spine_new> where { ?s rdfs:label ’NodeName’@kr. ?s rdf:type ?region2. ?s ?p ?o FILTER(regex(?region2, ’http://localhost:8891/spine#’))}
	select distinct ?s ?p ?o from <http://localhost:8890/spine_new> where { ?s rdfs:label ’NodeName’@kr. ?s rdf:type ?region2. ?s ?p ?o FILTER(regex(?region2, ’http://localhost:8891/spine#’))}

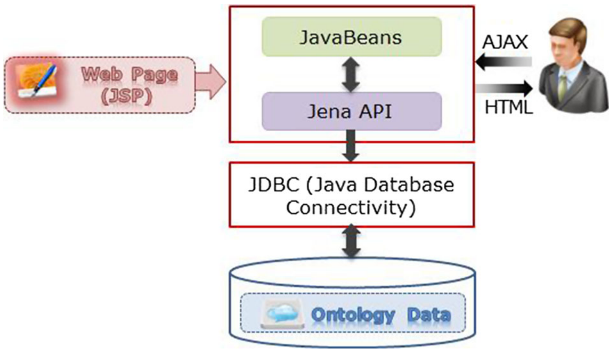

Fig. 3 Overall process of the ontology query module

developed by the non-profit Khronos Group [21] and can render interactive 3D computer graphics on a compatible Web browser. It is a cross-platform system based on Open Graphics Library for Embedded Systems (OpenGL ES) 2.0, and can be used by JavaScript without plug-ins. WebGL and ThreeJS are accessed via the HTML5 Canvas element using a Document Object Model interface [21,22]. The rendered 3D virtual spine can be moved, rotated, and zoomed through camera movements in response to a keyboard or mouse input. A certain part of the virtual spine can be selected and highlighted. Table 4 lists the user interfaces used for handling the 3D virtual human spine using a keyboard and a mouse.

Figure 4a shows an example of the 3D virtual human spine model visualization with the default position. When such visualization is conducted, the system brings the full list of spine bone internally by using Q2 query in Table 3. Figure 4b, c show movements of the model in the response to camera movements along the x- and y-axes, respectively. Figure 5a shows an example of model rotation using the keyboard. Figure 5b, c show camera translation and rotation using a mouse, respectively. Figure 5d shows an example of zooming into the model using a mouse. Figure 6a shows the spine

Table 4 User interfaces for manipulating the virtual spine

Device	User interface	Action
Keyboard	Arrow keys	Move a camera position (x-axis, y-axis)
	<, >	Rotate the virtual human spine
Mouse	Left button click	Select a component of the virtual spine
	Right button click	Rotate a camera position
	Wheel scroll	Zoom in/out
	Right button drag	Move a camera position

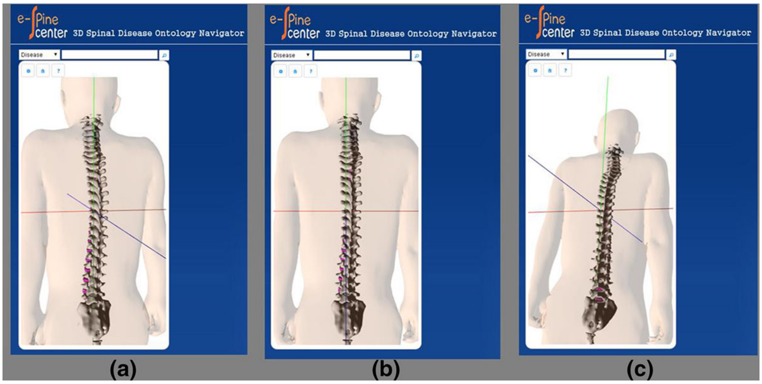

Fig. 4 **a** The 3D virtual spine model with the default position. **b** Translation of the camera position in the *x-axis* and **c** in the *y-axis* in response to keyboard input

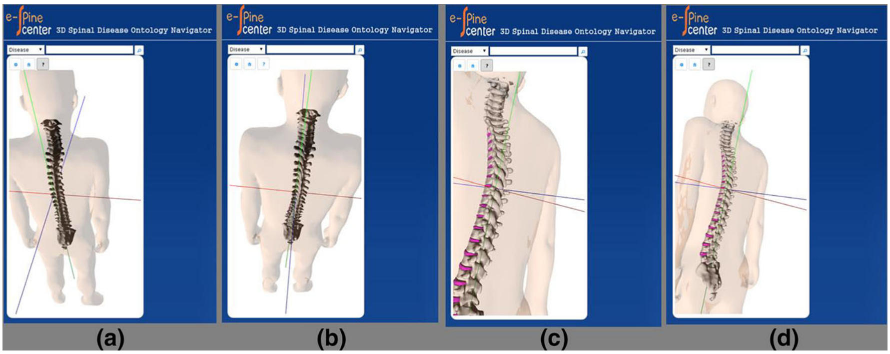

Fig. 5 **a** Rotation of the model using keyboard input. **b** Translation and **c** rotation of the camera position using mouse input, and **d** zooming into the model using a mouse

highlighted at the tenth thoracic vertebra (T10). Users can highlight the spinal column via a mouse click to view the column more accurately, showing its name. Users can also adjust the transparency of the area around the spine, as shown in Fig. 6.

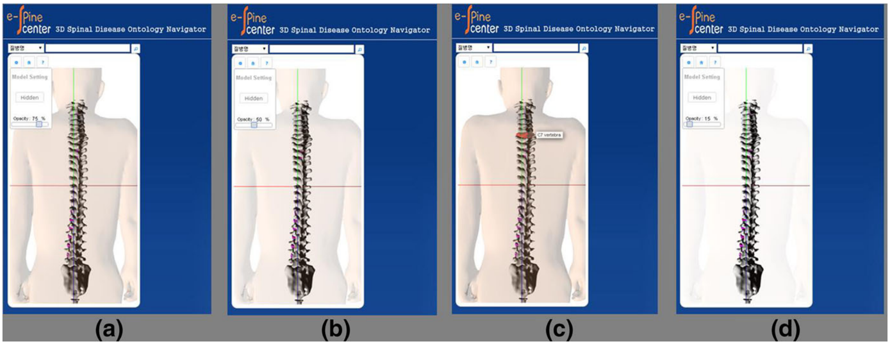

Fig. 6 The virtual spine highlighted at T10, **a** the model with the default transparency (50%), **b** with transparency of 75%, and **c** with transparency of 15%

3.4 Interactive visualization module

The interactive visualization module views retrieved ontology information by connecting it with the virtual human spine. It produces a dialog box in response to a disease search or a region search and displays the search results in the dialog box using AJAX, besides connecting the dialog box and the 3D virtual human spine model via a line using SVG.

Users can navigate specific spinal diseases by entering words in a textbox, and the ontological terms associated with the word are listed in a pull-down menu, as shown in Fig. 7a. For example, if the word 'spinal' is entered in the textbox, the navigation system will display the ontological terms that contain the word 'spinal' via a pull-down menu. Here, if a term is selected (e.g., 'spinal fracture') from the pull-down menu, the system will retrieve the ontology information associated with the selected term, such as the spinal diseases relevant to the term, together with their causes and treatments, which will be displayed in a popup dialog box, as shown in Fig. 7b. The retrieval is done by using Q3 with "PartName=T3(thoracic vertebra)" in Table 3, which can access list of diseases that occur at a specific site. The ontology data are provided with two viewing features, i.e., 'View All' and 'View by Category'. The default viewing feature is 'View All' which is used to visualize all items (diseases, symptoms, cause of the symptoms, code, relation information. etc) stored at a specific site, as shown in the dialog box in Fig. 7b. Figure 7c shows a results screen when selecting the 'View by Category' feature of them which they want to see. The dialog content is displayed dynamically in response to user input on the Web using AJAX.

Users can also navigate specific spinal diseases by selecting regions of the spine on the 3D virtual model by using the proposed query commands in Table 3. If a user selects a region on the model using mouse input, the system will show the ontological information associated with the selected region in a popup dialog box with two view features (i.e., 'View All' and 'View by Category'). To visualize the connectivity between the dialog box and the selected region of the model, we linked them with a line using SVG, as shown in Fig. 8. If the dialog box moves, the linked line will be redrawn by following the movements of the dialog box, as shown in Fig. 8b. Figure 8c

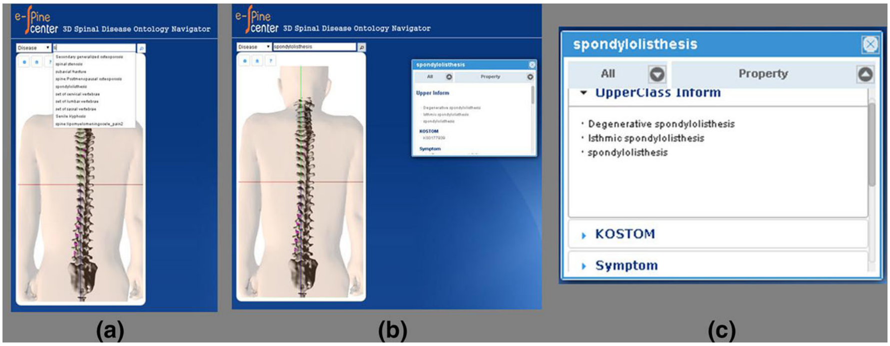

Fig. 7 **a** A pull-down menu listing the ontological terms associated with the user-entered word. **b** A dialog box showing the retrieved ontology information relevant to the selected term. **c** Viewing the ontology information using the View by Category feature

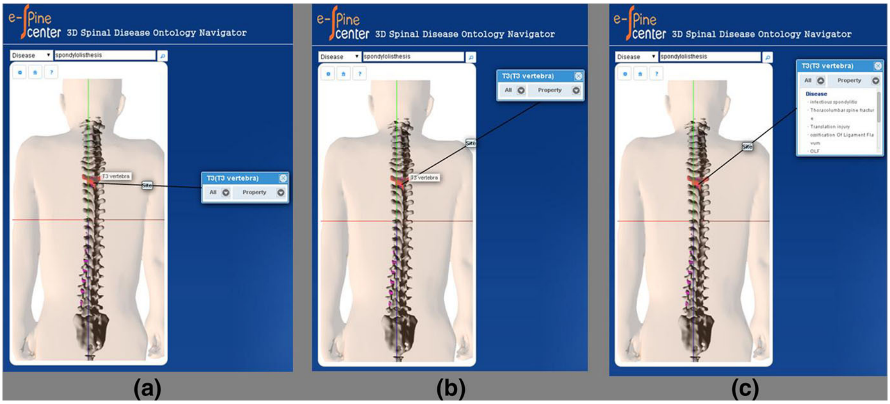

Fig. 8 **a** A dialog box displayed when selecting the T3 thoracic vertebra from the 3D model, **b** a result screen when moving this dialog box and **c** the ontology information displayed using the View All feature

shows the ontology information displayed using the 'View All' feature in the dialog box shown in Fig. 8a. The information includes the spinal diseases relevant to the selected region (i.e., the thoracic vertebra T3) and their causes and treatments.

If the user selects a disease from the dialog, the system will show detailed ontological information on the disease using a new disease dialog box, connecting it with the existing dialog box as shown in Fig. 9a. This information includes the definition and the locations of the diseases and relevant clinical diagnosis, causes, concomitant diseases, symptoms, and preoperative and postoperative images through query commands with the selected disease name. We provide the source images for the preoperative and postoperative images in a new window when the user selects them, as shown Fig. 9b.

If one of the disease locations (or regions) is clicked from the dialog box, the system will add a new dialog for the region by connecting it with the dialog box for the disease, as well as the 3D model, as shown in Fig. 10. Of course, such visualization of the disease information will be done through the related query commands.

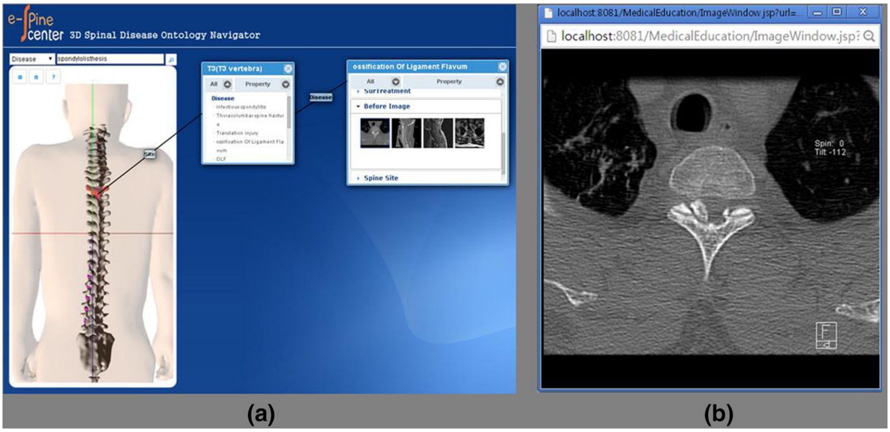

Fig. 9 **a** Viewing detailed ontology information for a selected disease in a new dialog box, and **b** a related preoperative source image [16, p. 278]

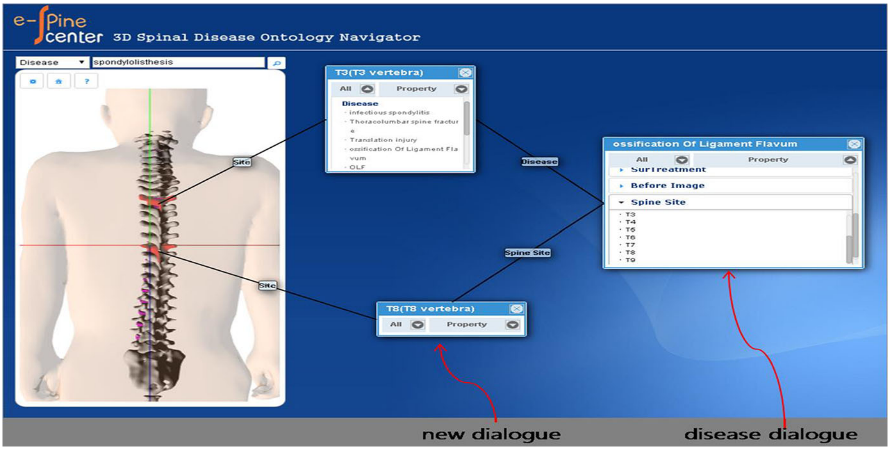

Fig. 10 An example showing an additional dialog box connected with the 3D model when the user selects a disease location in the disease dialog

When such a treatment is selected, the system will show these other diseases, together with detailed treatment plans using a new dialog box, as shown in Fig. 11.

The system is linked to the homepage of the Korean Standard Terminology of Medicine (KOSTOM) [24], which integrates all terms and concepts used in healthcare and the homepage of the Foundational Model of Anatomy ontology (FMA) [25], which is an evolving knowledge resource for biomedical informatics. This supports the synonyms and anatomical locations for the user-selected disease or region by providing access to further medical resources. The FMA is accessed through the Foundational Model Explorer (FME) developed for viewing the content of the FMA by the Structural Informatics Group at the University of Washington. Figure 12 shows examples of searching medical terms from the KOSTOM and FMA homepages.

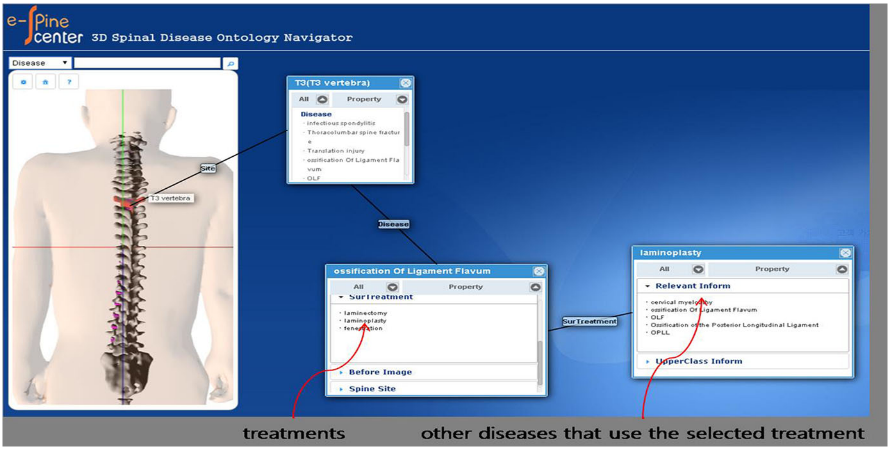

Fig. 11 Example showing the additional diseases that use the same treatment plan

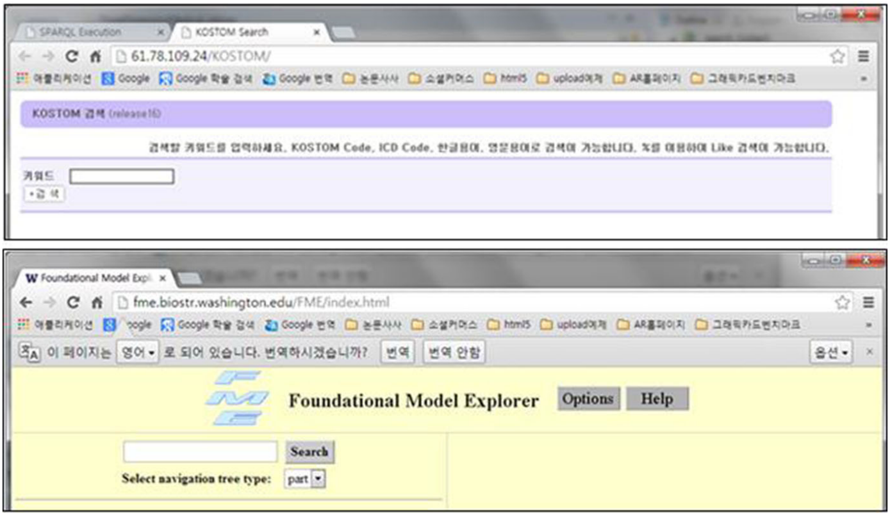

Fig. 12 Examples of searching medical terms from the KOSTOM (*top*) and FMA (*bottom*) homepages

4 Concluding remarks

A web-based 3D navigation system for the spinal disease ontology, which supports spinal specialists to simulate spinal disease via virtual experiments on a 3D model based on ontological interrelations, has been described. The system was implemented using Virtuosos SPARQL, Java, AJAX, HTML5, WebGL, and SVG on a Virtuoso server on the Web via JSP web pages.

The navigation system consists of three main modules: a query module, a 3D rendering module, and a visualization module. The query module retrieves spinal ontology data via web page. This module uploads the spinal disease ontology to the

Virtuoso server to manipulate the ontology data on the Web. It obtains ontological information via a JSP web page using Virtuosos SPARQL to query and analyze the data, i.e., all spinal diseases, spine components, specific spinal diseases that can occur in a particular region of the spine, or information of the parent–child nodes for a specific node in the ontology. The system then displays the query results using the Web. The 3D rendering module creates a virtual human spine model to enable users to simulate query results on it. The model can be moved and rotated via camera movements in response to mouse or keyboard input, and the transparency can be adjusted by the user. Specific locations in the virtual spine can be selected and highlighted, and the user can zoom in or out. The visualization module creates dialog boxes dynamically to display search results and connects these dialog boxes with the virtual spine model using lines.

Spine specialists can navigate specific spinal diseases by entering keywords in a textbox or selecting a region of the model. Search results include diseases relevant to the keywords or the selected region, together with their causes, definitions, locations, symptoms, treatments, images, and other relevant diseases. Spine specialists can simulate spinal diseases using the search results using the virtual spine in 3D with multiple viewing angles, which is expected to enable more accurate diagnoses of spinal diseases.

Although our web-based 3D ontology navigation system enables users in making accurate diagnoses of spinal diseases based on ontological information, a number of issues still remain. Practical trials via experiments in a clinical setting must be conducted to further evaluate the presented navigation system. Furthermore, we are going to evaluate our system through feedbacks of medical doctors.

Acknowledgements This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2014R1A1A2055379) and by the MSIP(Ministry of Science, ICT and Future Planning), Korea, under the ITRC(Information Technology Research Center) support program (IITP-2016-H8501-16-1013) supervised by the IITP(Institute for Information & communication Technology Promotion).

References

1. Kim GH, Kang MJ, Jung CY, Jung JY, Jung SE, Kim JS, Jeong JS, Kim DH, Yoo KH, Seo DM, Lee SW, Lee SB, Lee SH, Kim SK (2013) Development of Ontology for the Diseases of Spine. In: Jeon HY, Obaidat MS, Yen NY, Park JJ (eds.) CSA 2014. LNEE, 279:1171–1178. Springer, Heidelberg
2. Lee SB, Lee SH, Seo DM, Yoo K-H, Kim SK (2014) Development of ontology and 3D software for the diseases of spine. Adv Multimed 420848:1–4
3. Pfirrmann CWA, Metzdorf A, Zanetti M, Hodler J, Boos N (2001) Magnetic resonance classification of lumbar intervertebral disc degeneration. Spine 26(17):1873–1878
4. Modic MT, Ross JS (2007) Lumbar degenerative disk disease. Radiology 245(1):4361
5. Weishaupt D, Zanetti M, Boos N, Hodler J (1999) MR imaging and CT in osteoarthritis of the lumbar facet joints. Skelet Radiol 28(4):215–219
6. Koper R (2004) Use of the semantic web to solve some basic problems in education: increase flexible, distributed lifelong learning, decrease teachers workload. J Interact Media Educ 6:1–23 Special Issue on the Educational Semantic Web
7. Vaccaro AR, Koerner JD, Radcliff KE, Oner FC, Reinhold M, Schnake KJ, Kandziora F, Fehlings MG, Dvorak MF, Aarabi B, Rajasekaran S, Schroeder GD, Kepler CK, Vialle LR (2016) AOSpine subaxial cervical spine injury classification system. Eur Spine J 25(7):2173–2184

8. Dean M, Schreiber G(2004) OWL Web Ontology Language Reference. Technical report, World Wide Web Consortium (W3C)
9. SemanticWorks Semantic Web Tool, http://altova-semanticworks.soft112.com/, http://www.altova.com/documents/SemanticWorksdatasheet
10. Protege, http://protege.stanford.edu/
11. Neches R, Fikes R, Finin T, Gruber T, Patil R, Senator T, Swartout WR (1991) Enabling technology for knowledge sharing. AI Mag 12(3):36–56
12. Gruber TR (1993) Adaptive a translation approach to portable ontology specification. Knowl Acquis 5(2):199–220
13. Genesereth MR, Nilsson NJ (1987) Logical foundation of artificial intelligence. Morgan Kaufmann, Los Altos
14. Kim M, Choi SY (2010) An Ontology-based Adaptive Learning System to Enhance Self-directed Learning. In: Kang, B.H., Richard, D. (eds.) PKAW2010. LNCS, 6232:91–102. Springer, Heidelberg
15. Lee W, Leung Carson K-S, Lee JJ (2011) Mobile web navigation in digital ecosystems using rooted directed trees. IEEE Trans Ind Electron 58(6):2154–2162
16. Song MH, Jeong JS, Kim M, Lee SH, Choi SS, Choi ES, Doung C, Yoo KH (2015) An ontology navigation system for 3D spinal model. ACM BigDAS 2015. New York, USA, pp 277–279
17. SPARQL 1.1 Query Language. W3C Recommendation 21 March 2013, https://www.w3.org/TR/sparql11-query/
18. Virtuoso, http://en.wikipedia.org/wiki/Virtuoso_Universal_Server
19. VirtuosoUniversalServer, http://www.w3.org/wiki/VirtuosoUniversalServer
20. Scable Vector Graphics, https://www.w3.org/Graphics/SVG/
21. WebGL Specification, https://www.khronos.org/webgl/
22. Three.JS, http://www.threejs.org/
23. Stedmans Medical Dictionary, http://www.lww.com/stedmans
24. KOSTOM (Korean Standard Terminology of Medicine), http://61.78.109.24/KOSTOM/
25. FMA (Foundational Model of Anatomy Ontology), http://fme.biostr.washington.edu/FME/index.html