A Novel Semantic Annotation Algorithm for Models Based on Associated Scene

Jun Liu

School of Computer and Electronic Information, Guangxi University, Nanning, China
Email: liujunzky@163.com

Yan Liu*

Numerical Weather Prediction Center, China Meteorological Administration, Beijing, China
Email: liuyan@cma.gov.cn

Yu Sun, Wenzhen Su, Youhua Li and Zhenjin Pan

School of Computer and Electronic Information, Guangxi University, Nanning, China
Email: {sunyu1225, soovincent, youhua, zhenjinpan}@163.com

Abstract—**An automatic 3D semantic annotation algorithm is proposed based on the associated scene between models. This algorithm uses the semantic context of 3D models to annotate the non-annotated nodes in terms of the annotated nodes. Firstly, this algorithm abstracts the extensible 3D (X3D) scene into the X3D conventional scene tree, transforming the conventional scene tree into an in-line node scene tree. Additionally, the algorithm uses the X3D ontology for semantic reasoning between the scene and the models. Lastly, the algorithm annotates the model files automatically. This study analyses the time and space complexity of the algorithm. Experimental results confirm that this algorithm improves not only the annotation validity but also the retrieval performance.**

Index Terms—**scene, semantic annotation, X3D model, ontology**

I. INTRODUCTION

With the development of computer technology, 3D models have been widely used in education, network, design, entertainment and many other fields. Similarly, the number of 3D models has also grown rapidly. To find and use reusable 3D models effectively, instead of redesigning and regenerating 3D models, 3D model retrieval technology is researched. One of the current research hotspots is the semantic-based 3D model retrieval technology [1].

The semantic-based 3D model retrieval technology manages and accesses the 3D model data with the semantic concept to solve the semantic gap [2] between low-level features and high-level semantic concepts. The technology effectively improves the 3D model retrieval accuracy. However, problems in the semantic expression and usage of the semantic-based retrieval method still exist. The key to this problem lies in the semantic annotation [3, 4].

Aden [5] proposed a part-based annotation framework based on an ontological relation. He showed how the annotation framework might be used in two scenarios, namely, for the creation of avatars in emerging Internet-based virtual worlds and for product design in e-manufacturing. Goldfeder [6] used the models and annotated words in the semantic sample base to improve the recall of text search by automatically assigning appropriate tags to models. Ruan et al. [7] firstly built a semantic network by using WordNet [8] and Jena [9]. These authors presented the construction of a 3D model retrieval system on the basis of the ontology vocabularies. They attempted to create this method not only to effectively expand the semantic vocabularies of a 3D model library without abundant categories by enlarging the coverage of vocabularies but also to keep good semantic relevance of the expanded vocabularies to the original ones to achieve effective semantic-based 3D model retrieval. Yang [10] proposed an improved semantic automatic annotation algorithm. He combined a 3D model retrieval algorithm with annotated models. This algorithm was based on the visual orthogonal projection image with semantic similarity calculation. Huang [11] realized an improved semantic automatic annotation algorithm, which simultaneously considered the content feature correlation of models, annotation results of existing models in the model sample database and calculation rules based on WordNet semantic similarity.

Most of these automatic annotation algorithms were based on the semantic tree. Construction of the semantic tree to describe the semantic relationships of models in the model base allowed these algorithms to understand the semantics of the keywords with the aid of a common ontology knowledge base, such as WordNet. These

algorithms calculated the semantic relevance between keywords and models in the semantic tree nodes and thus made semantic annotation. These algorithms could partly improve the annotation results.

However, these annotation methods ignored the relationship between the scene and the models. In fact, the 3D models are not isolated in the scene. The 3D models interact with each other and form the entire scene. The relationship between the whole and the part plays an important role in the reasoning and retrieval process.

This paper presents an associated scene of 3D model semantic annotation algorithm. According to the associated scene of the 3D models and the ontology for semantic reasoning, the extensible 3D (X3D) files can be automatically annotated by using this algorithm.

II. X3D SCENE AND SEMANTIC ANNOTATION

Extensible 3D (X3D) [12] is a royalty-free open standard file format and run-time architecture that represents and communicates 3D scenes and objects over the web. X3D is an International Standards Organization standard that provides a system for the storage, retrieval, interaction and playback of real-time graphics content embedded in applications. In addition, X3D is a run-time architecture to represent and browse 3D scenes and objects by using extended markup language (XML). We select X3D as the 3D model format in our research. X3D scenes are environments that are based on the X3D format. Examples of X3D scenes include those for a tea plantation or a factory building.
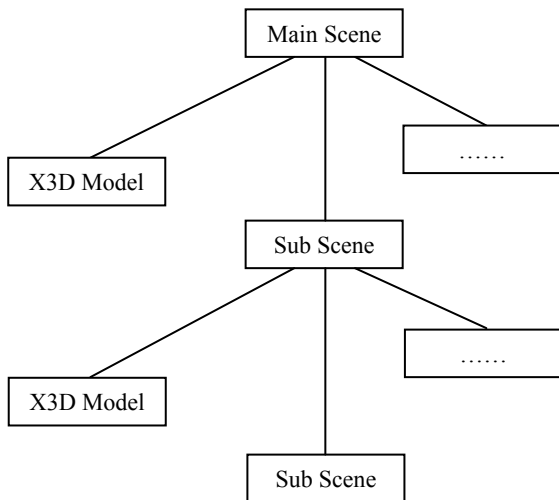

Figure 1. X3D Scene Tree

X3D metadata and structure include X3D nodes, XML tags, X3D document type declaration, header nodes, component nodes, metanodes and scene node. The scene node is the root node, which contains the X3D scene syntax definition. All nodes create a scene.

The X3D file can be considered as a main scene file. A main scene file can include multiple X3D models and multiple sub-scenes, as shown in Figure 1. Each node can be added and designed as an in-line node that points to another X3D file to achieve X3D program modular design. Therefore, in the design of complex scenes, the

main scene can be in-line with several sub-scenes. The sub-scenes can also be in-line with more sub-scenes. Therefore, with the in-line nodes, the X3D scene tree can be transformed into an in-line node scene tree, as shown in Figure 2.
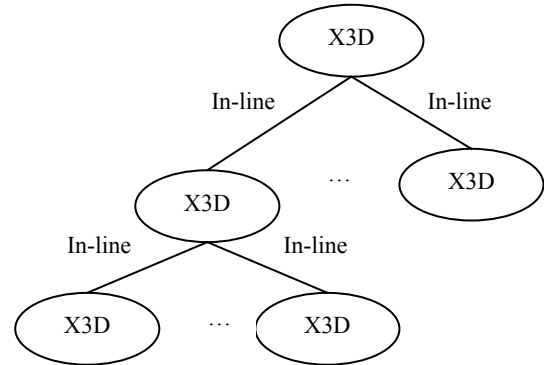

Figure 2. In-line Node Scene Tree

The term *ontology* has a variety of definitions [13, 14]. In 1997, Borst defined *ontology* as a 'formal specification of a shared conceptualization' [14]. We adopt this definition in this paper. Perez et al. researched on ontology classification, development and application [15, 16]. Ontology is introduced into the domain of annotation and retrieval of 3D model. On the one hand, ontology can intelligently guide the retrieval process on the basis of the semantic relationship between models. On the other hand, ontology can be used to connect the underlying characteristics with the high-level semantic information of models to achieve automatic semantic annotation. The traditional text- and content-based retrieval techniques had difficulties in achieving automatic semantic annotation. Thus, this paper presents a semantic annotation algorithm that uses semantic ontology reasoning for the automatic semantic annotation of X3D files.

III. ANNOSCENEBASED ALGORITHM

In our research, two constraints are observed: (1) some specific 3D models can be included only in some particular 3D scenes; 2) some unique 3D scenes can contain only some specific 3D models. These semantic constraints can be used for semantic annotation. Simultaneously, some semantic correlation between the 3D models exists in similar scenes.

We abstract the actual scene into the scene tree according to the model format shown in Figure 2. The semantic annotation of the nodes in the scene tree is to deduce the nodes, which will be annotated from the nodes that have been annotated. Two types of positional relationships are observed between the two nodes: directly connected and indirectly connected. Figures 3(1) to 3(3) show that the reasoning model includes three cases, in which the solid circles are the annotated nodes and the hollow circles are the non-annotated nodes.

- (1) The non-annotated node is located above the annotated node. In this case, direct reasoning about the father node can be performed from the

bottom up.

- (2) The annotated node is located above the non-annotated node. In this case, direct reasoning about the child node can be performed from the top down.
- (3) The non-annotated node and the annotated node are not connected directly and not necessarily in the same layer. In this case, the semantic annotation can be achieved by (1) several times and then by (2) several times.
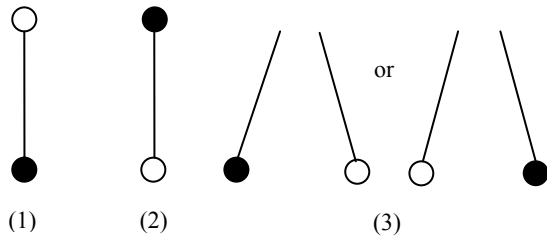

Figure 3. Three Cases of Reasoning

According to the characteristics of the 3D scene and the model format in our research, we propose a 3D model semantics automatic annotation algorithm called AnnoSceneBased.

Assume that $N_{current}$ is the current node set, $S_{current}$ is the semantic set, s_c is the semantic annotation set, the *ith* of the children of $N_{current}$ is called the $N_{child-i}$ (i is a positive integer), $S_{child-i}$ is the semantic set of the $N_{child-i}$, N_{father} is the father node of $N_{current}$ and the semantic set of the N_{father} is the S_{father}. Therefore, the AnnoSceneBased algorithm can be described as follows:

(1) First traverse:

(1.1) Create $S_{current}$ for $N_{current}$. If $N_{current}$ already has s_c, add s_c into $S_{current}$.

(1.2) If $N_{current}$ has child nodes, recursively call this algorithm with all its $N_{child-i}$ as the input.

(1.3) If $S_{current}$ is not empty, reason out the smallest semantic from $S_{current}$ to be the s_c of $N_{current}$. Else, return.

(1.4) If $N_{current}$ is not the root node, reason out s_f, which is the semantic of its father node. Add s_f into S_{father}, then return. Else, return.

(2) Second traverse:

(2.1) If $N_{current}$ already has s_c, add s_c into $S_{current}$.

(2.2) If $S_{current}$ is empty, return.

(2.3) If the length of $S_{current}$ is 1, set the element of $S_{current}$ as s_c.

(2.4) If the length of $S_{current}$ is larger than 1, reason out the smallest semantic from $S_{current}$ to be the s_c of $N_{current}$.

(2.5) If $N_{current}$ has child nodes, reason out s_c, which is the semantic of its child nodes. Add s_c into $S_{child-i}$ of all child nodes, then recursively call this algorithm with all its $N_{child-i}$ as the input. Else, return.

This algorithm must traverse the scene tree twice. For the first time, the algorithm annotates the father nodes from the annotated nodes to the root node. The smallest semantic of the collection, which is reasoned from the child nodes, will be the annotation name. After the first traverse, the direct or indirect father nodes of the annotated nodes are all annotated, that is, the nodes on

the paths from the annotated node to the root node are all annotated.

For the second time, the algorithm annotates the child nodes from the annotated nodes to the leaf nodes. The smallest semantic of the collection reasoned from the father node will be the annotation name. Thus far, the entire scene tree has been annotated.

Every node is traversed twice by this algorithm in the scene tree, as shown in (1.3), (1.4), (2.4) and (2.5). Each time, the complexity is O (n). Therefore, the total time complexity of this algorithm is O (n). An array is created for each node to store the semantic collection by this algorithm. The maximum length of the array is the number of its child nodes. Thus, the total space complexity of this algorithm is O (n).

IV. EXPERIMENTAL RESULTS AND ANALYSIS

Given the current 3D model's insufficiency of X3D scene and model format specification correlation, the retrieval evaluation set does not apply to this paper. To test the application results in the real environment, we use the existing 3D models and labels in the local model base of our software as the test set.

Protégé 4.1 [17] is used in our experiments as the ontology developing and modelling tool. The Java programming language is used to implement the algorithm to annotate the X3D files. The metanodes in the head section of the X3D format are responsible for describing the instructions. A metanode includes the name and the content domain. We will use the metanodes for X3D semantic annotation in this paper. The value of the name domain is 'scene' or 'model', indicating whether this X3D file is a scene or a model. The value of content domain is a specific category of this scene or model.

The following figures show the experimental data of 3D scene semantic annotation of some scenes and models. These experiments comprise three types of scenes: bedroom, parlour and restroom. Each scene has 30 scenes, leading to a total of 90 scenes. Without loss of generality, we select the curtain, bed and bathtub as the retrieved objects. The curtains can be positioned in three types of scenes, and each scene can include any number of curtains. The beds can be placed only in the bedroom, and each bedroom can include any number of beds. Bathtubs can be placed only in the restroom, and each restroom can include only one bathtub. A total of 100 curtains are positioned in 90 scenes randomly; a total of 50 beds are enclosed in bedrooms randomly; a total of 30 bathtubs are enclosed in restrooms. Moreover, 20 wardrobes are placed in bedrooms, 20 sofas are placed in parlours and 20 washers are placed in the restroom. All objects are positioned in the scene at random. The wardrobes, sofas and washers are employed in the reasoning of curtains, beds and bathtubs. We call them assistance models in this paper.

We use *Precision* and *Recall* as evaluation parameters. The definitions of *Precision* and *Recall* are as follows:

$$Precision = N_c/N_r \qquad (1)$$
$$Recall = N_c/N_a \qquad (2)$$

where N_c is the number of correctly retrieved models, N_r is the total number of retrieved models and N_a is the number of correct models in the entire model base.

Figure 4 shows the object characteristic management interface of 3D models in the local model base of our software as the test set. The annotated semantic context of 3D models from the AnnoSceneBased algorithm can be browsed in the object characteristic management interface.
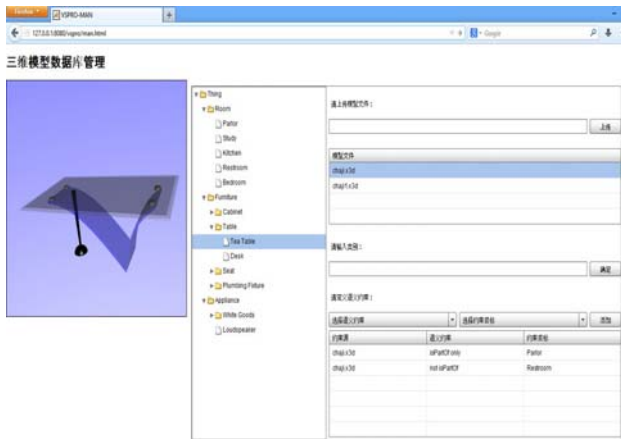

Figure 4. 3D Model Characteristics Management Interface

Figure 5 shows the 3D model retrieval results from the annotated 3D models, which are annotated by the AnnoSceneBased algorithm.
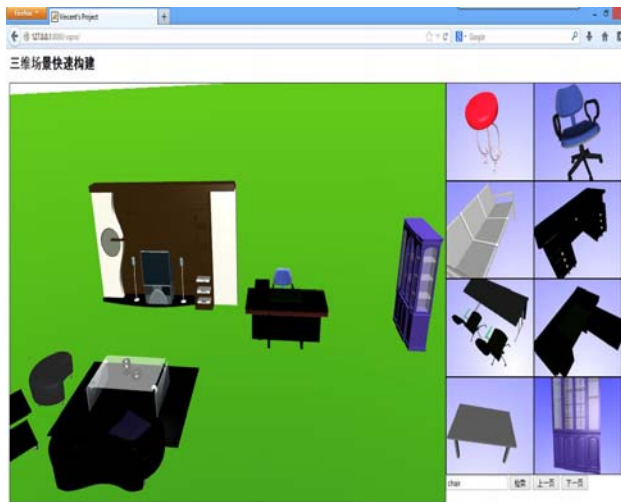

Figure 5. 3D Model Retrieval Results

Figures 6 to 10 show the precision and recall curves of our experiments. Figure 6 corresponds to this occasion: the scenes and retrieval objects are not annotated, and the annotated rate of assistance models is increased progressively (occasion 1).

Given that the curtains do not belong to any distinct scene, their semantic cannot be reasoned precisely by the models in the same scene, whereas the semantic of the beds and bathtubs can be reasoned. The precisions of the bed and bathtub do not improve with the increase of the annotated rate of assistance models. The reason for this phenomenon is that semantic reasoning can only deduce

that the curtains belong to a distinctive scene, but not their real model category. However, the recalls improve. This improvement verifies the semantic relevance of the models in the same scenes.
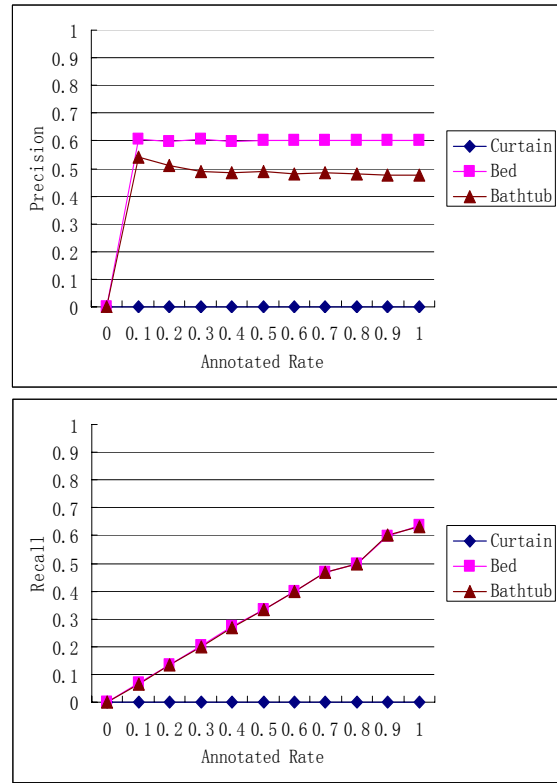

Figure 6. AnnoSceneBased Algorithm on Occasion 1

Figures 7 and 8 correspond to this occasion: the scenes are not annotated; the retrieved objects are 30% annotated; and the annotated rate of assistance models is increased progressively (occasion 2). To contrast the traditional WordNet retrieval method [13], this occasion presents a little change. Figure 7 shows the results of the traditional WordNet retrieval method, and Figure 8 shows the results of the AnnoSceneBased algorithm.

In the traditional WordNet retrieval method, changing the annotated rate of assistance models will not have any effect on the annotation of retrieved objects. Therefore, the precision and recall are gentle. In our AnnoSceneBased algorithm, the precision and recall are greater than those in the traditional method when the annotated rate of assistance models increases because of the effect of semantic reasoning about assistance models. However, the curtains are special cases because they do not belong to any special scene. They are annotated as the items on special scene after semantic reasoning. For this reason, the recall of the curtain decreases. This decrease verifies the superiority of our AnnoSceneBased algorithm over the traditional WordNet retrieval method.

Figure 9 corresponds to this occasion: the assistance models are not annotated; the retrieved objects are 30% annotated; and the annotated rate of scenes is increased progressively (occasion 3). The precisions of bed and bathtub change minimally, but their recalls increase when the annotated rate of scenes increases. The curtains are

also special cases. Given that the curtains do not belong to any special scene, the precision increases. However, given that they are also annotated as items on the special scene, the recall decreases. This decrease verifies the semantic relevance between the scene and the models in it.

not annotated; the annotated rate of models is increased progressively; and the retrieved objects include the bedroom, parlour and restroom (occasion 4). This occasion verifies the feasibility of retrieval for multiple scenes. By using the AnnoSceneBased algorithm, we can annotate not only the models but also the scenes.
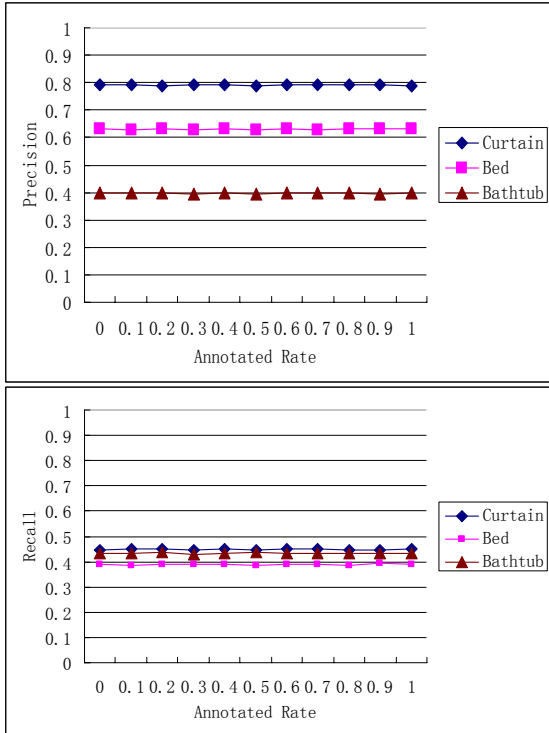

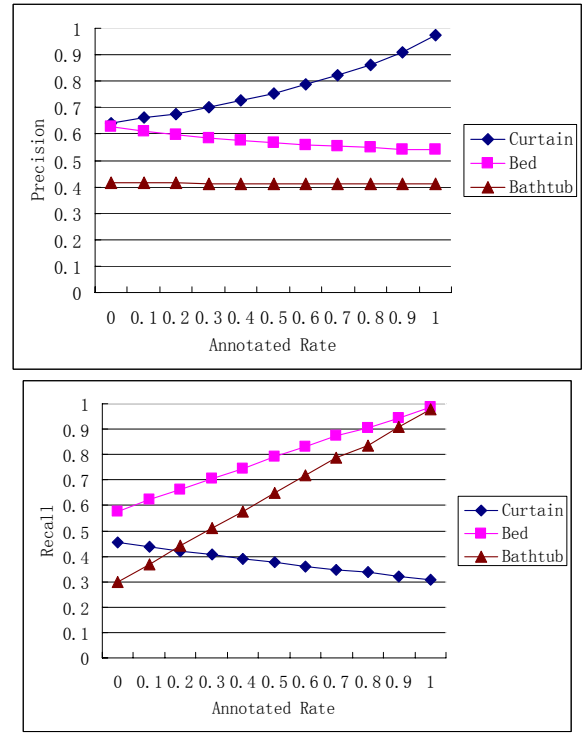Figure 7. WordNet Method on Occasion 2
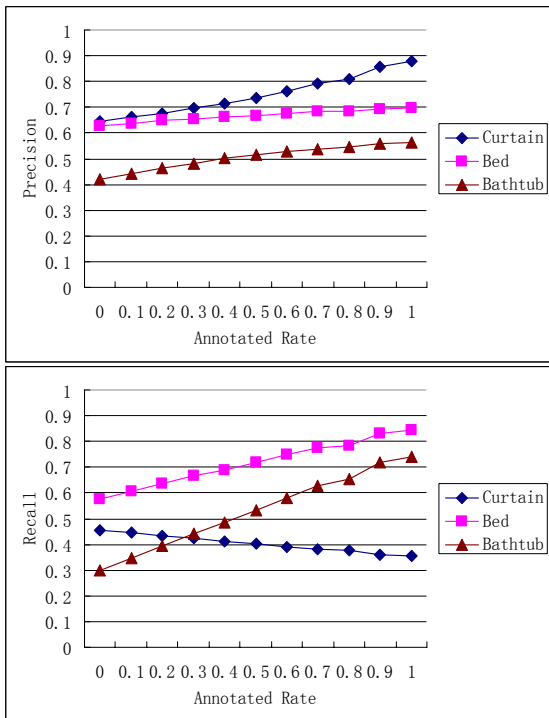

Figure9. AnnoSceneBased Algorithm on Occasion 3

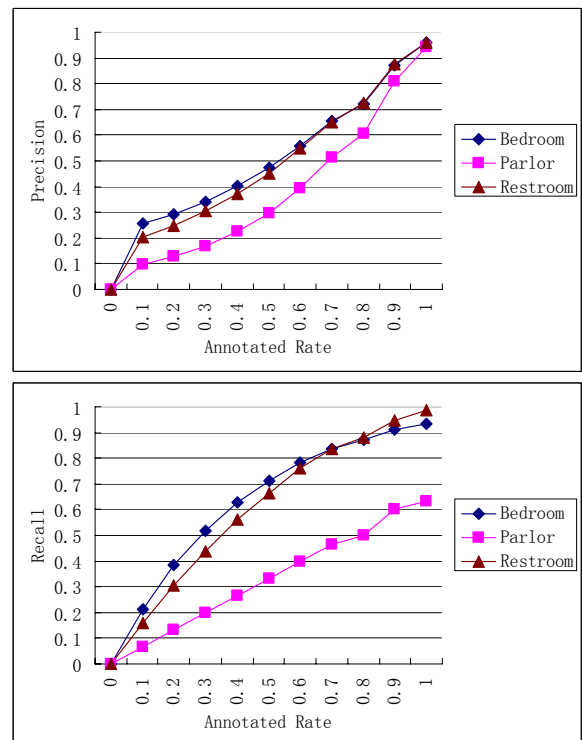

Figure8. AnnoSceneBased Algorithm on Occasion 2

Figure10. AnnoSceneBased Algorithm on Occasion 4

Figure 10 corresponds to this occasion: the scenes are

V. CONCLUSION

In this paper, the scene tree is firstly transformed into an in-line node scene tree. Subsequently, a 3D model semantic annotation algorithm is designed and achieved according to the characteristics of the X3D file format and the ontology. This semantic annotation algorithm uses the ontology for semantic reasoning between the scene and the models and annotates model files automatically. By using the AnnoSceneBased algorithm, we can automatically annotate the 3D models with the semantic contents, which reflect the corresponding characteristics. Available semantic guidance information leads to better retrieval performance. We will apply the associated scene 3D model semantic annotation to the 3D scene construction in the next stage.

ACKNOWLEDGMENT

REFERENCES

[1] PAN Xiang, ZHANG San-Yuan, YE Xiu-Zi. A Survey of Content-Based 3D Model Retrieval with Semantic Features [J]. Chinese Journal of Computers, 2009, 32(6): 1069-1079.

[2] Benjamin Bustos, etc. Automatic Selection and Combination of Descriptors for Effective 3D Similarity Search. IEEE Sixth International Symposium on Multimedia Software Engineering. 2004 [C], 514-521.

[3] Tianyang Lv. The researches on the 3D model retrieval techniques based on clustering analysis and based on semantic [D]. Changchun: Jilin University, 2007.

[4] TIAN Feng, SHEN Xu-kun, LIU Xian-mei et al. 3D Model Semantic Automatic Annotation Based on Weak Label [J]. Journal of System Simulation, 2012, 24(9): 1873-1881.

[5] Attene M, Robbiano F, Spagnuolo M, et al. Part-based Annotation of Virtual 3D Shapes. 2007 International Conference on Cyberworlds, 2007: 427-436.

[6] Goldfeder C, Allen P. Autotagging to improve text search for 3d models [C]. Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries. ACM, 2008: 355-358.

[7] JiabinRuan, YubinYang, JinJie Ruan, et al. Semantic based 3D Model Retrieval Using Ontology [J]. Computer Science, 2009, 36(2): 153-185.

[8] C. Felbaum. WordNet: an Electronic Lexical Database [M]. MIT Press, Cambridge, Massachusetts, 1998.

[9] HP Labs Semantic Web Research. Jena2A Semantic Web Framework for Java [EB/OL]. http://jena-sourceforge.net/ index.html, 2008-05-14

[10] Shaobo Yang. Research on Visual image-based 3D Model Retrieval and Semantic Automatic Annotation Technology [D]. Xian: Northwest University, 2010.

[11] Wen Huang. Design and Implementation of 3D Models [D]. Xian: Northwest University, 2011.

[12] WEB3D CONSORTIUM. Extensible 3D (X3D) Graphics specification. ISO/IEC 19775: 2004. http://www.web3d. org/x3d/specifications/x3d.

[13] Gruber T R. A Translation Approach to Portable Ontology Specifications [J]. Knowledge Acquisition, 1993, 5(2): 199-220.

[14] Borst W. N. Construction of engineering ontologies for knowledge sharing and reuse. PH. D. Thesis. University of Twente. 1997.

[15] A. G. Perez, V. R. Benjamins. Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem Solving Methods. Proceedings of the IJCAI- 99 workshop on Ontologies and Problem Solving Methods, 1999: 1-15.

[16] Noy N F, McGuinness D L. Ontology development 101: a guide to creating your first ontology. SMI Technical Report, SMI-2001-0880, Stanford University, USA.

[17] Protégé [EB/OL] [2009-3-16] http://protege.stanford.edu/.

Jun Liu, is an associate professor in the School of Computer and Electronic Information of Guangxi University, Nanning, China. His research interests include computer simulation, parallel and distributed computing and software engineering. He received a PhD in Computer Engineering from the Institute of Software, Chinese Academy of Science. You may contact him at liujunzky@163.com.

Yan Liu, is an associate professor in the Numerical Weather Prediction Center, China Meteorological Administration, Beijing, China. Her research interests include data assimilation and simulation. She received a PhD from Chinese Academy of Science. Contact her at liuyan@cma.gov.cn.

Yu Sun, is a lecturer in the School of Computer and Electronic Information of Guangxi University, Nanning, China. Her research interests include image processing and artificial intelligence. She received a PhD from Wuhan University. Contact her at sunyu1225@163.com.

Wenzhen Su, is a postgraduate in the School of Computer and Electronic Information of Guangxi University, Nanning, China. His research interests include computer simulation and virtual reality. Contact him at soovincent@163.com.

Youhua Li, is a postgraduate in the School of Computer and Electronic Information of Guangxi University, Nanning, China. His research interests include computer simulation and virtual reality. Contact him at youhua@163.com.

Zhenjin Pan, is a lecturer of computer department in Guangxi Second Light Industry Technical School, Nanning, China. She graduated from Kunming University of Science & Technology. She is a postgraduate in the School of Computer and Electronic Information of Guangxi University at present. Her research interests include computer simulation and virtual reality. Contact her at zhenjinpan@163.com.