Creating Meaningful Environment Models for Augmented Reality

Eduardo E. Veas∗

Dieter Schmalstieg

Technische Universität Graz

ABSTRACT

This article introduces a framework to generate three-dimensional models for Augmented Reality (AR) including semantics. The semantics of 3D models have been studied previously in the field of intelligent virtual environments, but the process of creating real-world models containing such information have received little attention. We introduce a method for the creation of semantic, 3D models of the environment using an AR application named InventAry. Assisted by an ontology, InventAry enforces the creation of combined geometric and semantic environment model.

1 INTRODUCTION

Semantic representations for 3D have been the focus of various studies in Virtual Reality (VR). Luck and Aylett [1] aimed at incorporating intelligent agents in virtual environments, introducing the concept of Intelligent Virtual Environment (IVE). Latoschick et al. [4] presented a symbolic representation to capture semantic and geometric knowledge. Kalogerakis et al. [3] suggested an ontological representation combining knowledge of different domains and VR scenes. With the exception of [3], semantics has been treated mostly as a separate asset of environment models in the past, requiring a separate annotation process. In most cases, Augmented Reality (AR) applications are limited to interact with a few physical objects, and their meaning is an implicit part of the application logic. As AR systems attempt to integrate activities dispersed in wider areas they require additional information about the surrounding world, and the mechanisms to manage it. Little work has addressed modelling non-geometric aspects. Newman et. al [5], describes a relational database incorporating geometric and semantic aspects of an office building and an AR tool to update it. However, the semantic interpretation of the relational data is left to the application. Höllerer et al. [2] uses a semantic database to infer topological information relevant for a mobile AR user, but the authors do not describe how to create this database. To the best of our knowledge, no AR modelling system specifically aimed at creating semantic information exists to date.

Our approach considers both geometric and semantic attributes when creating the environment model. An ontology assists the workflow by categorizing objects that may be found in the environment and how they relate. Whereas these are defined in advance using an ontology editor, instances representing concrete entities and their relations are created in realtime with a mobile AR application called InventAry. Furthermore, topological restrictions derived from semantic relations are used to constrain user interface operations, simplifying spatial interaction.

2 SEMANTICS FOR AUGMENTED REALITY

In our approach, knowledge has been distributed across interconnected ontologies, represented in OWL format, emphasizing modularity and extensibility (Figure 1). Our ontology bridges two views of space, the strictly geometrical and the topological, enabling applications to consider both aspects of reality. A geometry ontology defines transformations, coordinate frames, shapes, and relations to express that a certain shape refers to some coordinate frame. An ontology describing space and the entities that occupy space defines general concepts subsuming mobile and fixed entities and the central SpatialThing, SpatialThing and its subclasses (SpatialRegion and PhysicalObject) define operations of region connection calculus, useful to derive topological information about regions (i.e. when are regions overlapping, connected, etc.).

Extension ontologies incorporate domain knowledge, defining new entities and relating them to core spatial ontologies. Domains of interest for AR applications include tracking, sensors and activity locations. Over 80 concepts have been implemented, many of which are still being extended as new requirements become evident.

3 INVENTARY: MODELLING WITH SEMANTICS

A mobile AR application is used to create a semantic 3D model of the environment while exploring. The ontological description of the world assists the user while creating a knowledge base of it. The main activity starts by identifying a potential object as belonging to some category from the ontology. A graph displays the available categories, once an object is recognized, it can be instantiated. InventAry changes then to edit mode, in which class restrictions are used to identify possible relations and “highlight” targets for them directly in the scene. For example, instances of Door can only be placed on instances of Wall. InventAry uses such feature, common in an ontology editor, to simplify interaction in real-time.

InventAry supports primitive transformations for object positioning and manipulation (rotation, translation and scaling). However, such operations are time-consuming, particularly in a mobile setup. Therefore, InventAry confines them to topological relations (Figure 2). Using such relations as constraints reduces the degrees of freedom needed for interaction. The approach resembles the one presented by Stürzlinger [6] with the difference that InventAry does not require offline labeling every object explicitly with the possi-
able constraints, but allows in situ selection of constraints based on the location ontology. Examples of topological relations are:

\text{on}(i, j): i \text{ refers to } j, \text{ placement has to be on a surface of } j. \text{ Thus, when manipulating } i, \text{ the system forces it to stay on the surface of } j. \text{ An object restricted to be on another, has a maximum of } 3 \text{ (2 translational + 1 rotational) degrees of freedom.}

\text{onTopOf}(i, j): i \text{ refers to } j \text{ and the } +y \text{ component of the frame of reference of } i \text{ is restricted to the } +y \text{ component of the frame of reference of } j. \text{ It is a particular case of on, where the surface to which objects are restricted is the top surface.}

\text{against}(i, j): i \text{ refers to } j, \text{ such that on}(i, j) \text{ but not onTopOf}(i, j). \text{ That is, it refers } i \text{ to one of the vertical surfaces of } j. \text{ Also restricting movement to three degrees of freedom.}

\text{hangingFrom}(i, j): i \text{ refers to } j, \text{ and its } y \text{ component is restricted to the } -y \text{ component of } j. \text{ } i \text{ can be moved in the } y \text{ axis, as long as it does not colide with } j.

\text{Figure 2: Scenegraphs of topological relations. Constraint enforcing engines (CEE) connect the objects involved in such relations, restricting the degrees of freedom for manipulation. The left scene shows the relation between wall3 and ARToolKitMarker9. The right one shows objects related to wall3, and floor2. The screen-capture shows possible directions of movement of the objects involved.}

\text{Combining location relations such as the ones defined above with class restrictions, it becomes much easier to position an instance of Door (an on relation applied to Doors can only target Wall instances). After creating a Door, the on property is selected. Semantics help find and highlight entities that could be related through that property, namely Walls. By selecting one of them, all geometric operations are restricted to its surface, reducing the translations and rotations needed to position door1 accurately in space.}

\text{Once the instance satisfies the user’s intention at modelling reality it is added to the environment model. This may trigger subsequent tasks depending on the ontology and whether relations that the user did not consider can be derived. A reasoner infers the existence of new entities, as the model is created. For example, when adding an instance of Projector and an instance of Screen, the fact that a DisplayDevice is available in the area can be derived. A reasoner can also derive relations between entities. Stating that some physical object isTrackedBy a tracker (e.g. an instance of ARToolKitMarker) will add a refersTo relation to the tracker, associating thus the object’s shape to the its transform.)

\text{4 IMPLEMENTATION}

\text{InventAry is implemented as an application of the Studierstube framework, using the Pellet reasoner for validation. A proof-of-concept scenario has been implemented using an outside-in infrared tracking system in an indoor setup. We expect to extend it to a wide area coverage, hybrid tracking system. Special scenegraph nodes have been developed to serve as repositories for semantic entities. Communication with the reasoner is too expensive, preventing real-time response in matters related to geometric transformations during interaction. Therefore, the input that modifies an object in 3D space is captured by a constraint enforcing engine (i.e., functional constraints in the scene graph), which analyses the relations of the given object and corrects the positioning accordingly.)

\text{5 CONCLUSIONS AND FUTURE WORK}

\text{This article takes a first step towards the systematic creation of semantic annotations using AR applications. It provides a classification for the entities available, including a proper taxonomic organization. Besides, by combining topological relations with geometric restrictions, spatial interaction with objects is reduced to relating them to already existing objects and then using constraint-guided spatial interaction for the final placement. The resulting model needs no longer be annotated separately with semantic markup. Semantics are readily accessible to be exploited by other applications. For example, AR information systems such as [2] can apply visualization strategies based on semantic filtering.)

\text{The ontologies presented are being extended to address new issues, and new ways to support the modelling task. Another goal of InventAry is to assist the creation of tracking model through inference. For example, registration procedures can be aided by using geometric constraints. Sensor fusion configurations can be automatically derived from the existence of certain sensor patterns, for which ontology-based inference may provide a suitable tool.)

\text{ACKNOWLEDGEMENTS}

\text{This work was sponsored by the Austrian Science Fund FWF under contract Y193.)

\text{REFERENCES}


\text{[2] T. Höllerer, H. Navdeep, and T. Steven. Steps toward accommodating variable position tracking accuracy in a mobile augmented reality system, 2001.}


\text{http://www.studierstube.org}