Semantic Annotation of Digital 3D Objects

Marco Attene, Francesco Robbiano, Giuseppe Patanè,

Michela Mortara, Michela Spagnuolo and Bianca Falcidieno¹

Abstract-In this demo we show two new techniques for performing semantic annotation of 3D digital objects. The first tool, the ShapeAnnotator, is more general-purpose, and concerns a system to perform non-trivial segmentations of 3D surface meshes and to annotate the detected parts through concepts expressed by an ontology. Each part is eventually connected to an instance in a knowledge base to ease the retrieval process in a semantics-based context. The second approach is more contextspecific, as it is tailored on the domain of Virtual Humans; therefore, the conceptual tags are completely and uniquely defined by the domain of expertise. Both approaches share the goal of a semantic-driven annotation that considers the structure and the subparts of the original digital objects. Climbing the semantic ladder, i.e. passing from geometrical to structural to semantic description of objects, is one of the main goals of the AIM@SHAPE Network of Excellence [1]. Obviously, the more specific is the domain, the more the annotation process can be automated.

I. INTRODUCTION

 \mathbf{I}_n the latest years we have assisted to an impressive growth of online repositories of 3D shapes which reveals the importance of making these resources more accessible and easy to share and retrieve. Thus, characterizing shapes of a given domain is becoming more and more important, and specific annotation approaches that require minimal human intervention must be devised. To achieve this goal, the structural subdivision of an object into subparts, or segments, has proven to be a key issue. At a cognitive level, in fact, the comprehension of an object is often achieved by understanding its subparts. Therefore, the retrieval of 3D objects within a repository can be significantly improved by annotating each shape not only as a whole, but also in terms of its meaningful subparts, their attributes and their mutual relations. In general, both the extraction and the annotation of the subparts are characterized by an inherent context dependence: the kind of geometric analysis used to detect the segments, as well as the interpretation of the segments, can significantly vary in different contexts. A nearly-cylindrical object can be annotated as a finger in the domain of human bodies, as a piston in the domain of car engines, and may be

not detected at all in another domain in which this kind of features is not interesting.

Generally speaking, the purpose of annotation is to create correspondences between objects, or segments, and conceptual tags. Once an object and/or its parts are annotated, they can easily match textual searches. Stated differently, advanced and semantics-based annotation mechanisms support content-based retrieval within the framework of standard textual search engines.

II. THE SHAPEANNOTATOR

In the case of 3D shapes, different segmentation algorithms are available, but each of them aims at subdividing the shape considering specific strategies. It is easy to observe that, due to intrinsic limitations, no single algorithm can be used to provide rich segmentations which are meaningful in *any* context, even within a single domain. This motivates the introduction of a theoretical framework for working with multi-segmentations, which allow for a much more flexible support for semantic segmentation. The intuition behind multi-segmentation is that a meaningful shape segmentation is obtained by using in parallel a set of segmentation algorithms and by selecting and refining the detected segments. Some of these segments can be "elected" and tagged with concepts from an ontology. This step is addressed to as *annotation*.

We defined an annotation pipeline and developed a prototype graphical tool called the *ShapeAnnotator* [3]. This tool has been specifically designed to assist an expert user in the task of annotating a surface mesh with semantics belonging to a domain of expertise[2].

After loading a model and a domain ontology, the first step of the annotation pipeline is the feature identification, i.e. the execution of segmentation algorithms to build the multisegmented mesh. Once done, from the resulting multisegmented mesh interesting features are interactively selected. Each interesting feature is then annotated by creating an instance of a concept described in the ontology. Optionally, the system may be also programmed to automatically compute attributes and relations among the instances to significantly enrich the resulting knowledge base.

In order to identify surface features, the ShapeAnnotator provides a set of segmentation algorithms. Our prototype has a plugin-based architecture so that it is possible to import proper segmentation algorithms according to the requirements of the specific class of features. In the current implementation, we

¹ All the authors belong to the Istituto di Matematica Applicata e Tecnologie Informatiche – Via De Marini, 6, 16149 Genova, CNR, Italy. E-mail{attene,robbiano,patane,mortara,spagnuolo,falcidieno}@ge.imati.cnr.it.

have chosen a number of algorithms that cover quite a wide range of feature types. In particular, it is possible to capture planar features, generalized tubular features, primitive shapes such as planes, spheres and cylinders and protrusions.

To annotate the features, the user may select proper conceptual tags within a domain of expertise formalized as an OWL ontology. Once a proper concept has been identified, the ShapeAnnotator provides the possibility to create an instance, which means providing a URI (i.e., a unique name) and setting the value of the properties (attributes and relations) defined in the ontology for the class being instantiated.

Currently, our system requires the user to manually select the concepts to instantiate; for attributes and relations between instances, however, there is the possibility to tell the ShapeAnnotator how these properties can be calculated without the user intervention. The ShapeAnnotator, in fact, comes with a set of functionalities to measure geometric aspects of shape parts (i.e. bounding box length, radius of best-fitting cylinder, ...) and to establish topological relations among the parts (i.e. adjacency, containment, overlap, ...). We these functionalities segmentmeters. segmentmeters work independently of any ontology, the user may define their interpretation within each specific domain of annotation. Namely, the user may establish a set of connections between topological relations and conceptual relations (e.g. "segment adjacency" and is connected to) and between calculated values and class attributes (e.g. "radius of best-fitting cylinder" and through hole::radius). After having established such connections, the instance properties are transparently computed by the system. Furthermore, there is the possibility to combine some segmentmeters within formulae to be connected to specific attributes. Since we believe that modularity is crucial to provide a flexible annotation framework, we made our system able to load additional segmentmeters implemented externally as plug-ins, just as we have done for the segmentation algorithms.

The result of the annotation process is a set of instances that, together with the domain ontology, form a knowledge base.

III. SEMANTIC ANNOTATION OF VIRTUAL HUMANS

Decomposing an object into parts having a precise morphological meaning, such as tubular parts, has a deep impact in the classification of articulated shapes. Such a morphological segmentation may be expressive enough to allow an *automatic annotation* of components with semantic content, in well specified context domains, like that of *human body models* [4]. In fact, while geometric attributes may vary a lot from a model to another, the human body structure is well defined and the basic components are predominantly tubular (arms, legs, fingers, neck). To guarantee a right annotation, we admit that arms and legs are in any position but do not touch other body parts; moreover, we require that no other tubular shaped object touches the body (e.g. a man holding a stick). In this case, the segmentation would generate

tubular parts that the annotator is not able to label. The annotation can be defined as a function from a set of segments into a set of labels. In the specific, segments are those given by *Plumber*, i.e. may be either tubes or bodies. The labels are defined in order to make the annotation exhaustive with respect to the Plumber segmentation. The set L of labels is therefore the following: L={trunk, arm, hand, palm, finger, fingertip, leg, foot, neck, head}.

In general, not all the labels appear in the same annotation: for instance, fingers might be not segmented because of the posture or the poor quality of the scans. Hence, the hand segment will be labelled as hand, discarding the palm, finger and fingertip labels. Conversely, fingers, fingertips and palm will be instantiated at the expense of hand, unless deduced afterwards that adjacent regions labelled as palm, finger and fingertip form a hand. If the aforementioned requirements are met, then all the segments are labelled. The annotator exploits the geometric attributes of parts, computed during the segmentation phase, i.e., axis length and maximum, minimum and average length of cross sections; volume for bodies. Moreover the adjacency relations among segments coded in the shape graph are intensively employed. We point out that a tube segment has always two adjacent segments, while a body segment may be adjacent to one or more parts. In particular, we will call cap a body segment adjacent to one segment exactly.

IV. CONCLUSIONS

The paper has described two approaches that provide a semantic-driven annotation based on the structure and the subparts of the original digital objects. The general framework provided by the ShapeAnnotator allows a human expert to transform his/her knowledge about a shape into a formal and explicit content. By using shared ontologies, this content can be understood and exploited by other users and by software agents. In the specific domain of virtual humans, the Plumber algorithm automatically performs the shape annotation. The generalization of this approach to a broader set of domains is foreseen as future work.

ACKNOWLEDGMENT

This work is supported by the AIM@SHAPE Network of Excellence (EC FP6 IST Contract N. 506766).

REFERENCES

- [1] Network of Excellence AIM@SHAPE, EU FP6 IST NoE 506766. Available: http://www.aimatshape.net
- [2] Attene, M., Robbiano, F., Spagnuolo, M., Falcidieno, B. (2007) Part-based Annotation of Virtual 3D Shapes. In Proceedings of Cyberworlds '07, spec. sess. on NASAGEM workshop, IEEE Computer Society Press, pp. 427-436
- [3] The ShapeAnnotator. Available: http://shapeannotator.sourceforge.net
- [4] Dellas, F., Moccozet, L., Magnenat-Thalmann, N., Patané, G., Mortara, M. Spagnuolo, M., Falcidieno, B. (2007) Knowledge-based extraction of control skeleton for animation. In Proceedings of Shape Modeling International '07. IEEE Computer Society Press, pp. 51-60