The Role of Semantics in Shape Modelling and Reasoning

B. Falcidieno, M. Attene, M. Mortara

Istituto di Matematica Applicata e Tecnologie Informatiche
Consiglio Nazionale delle Ricerche
Genova, Italy
{bianca.falcidieno, marco.attene, michela.mortara}@ge.imati.cnr.it

1 Abstract

Due to recent technological advances, 3D geometry is becoming commonplace on all the entry-level PCs connected to the Internet, and multimedia-enabled systems must be prepared to a great increase of complex 3D content in the future. However, while Computer Graphics research has solved many problems related to the creation and manipulation of digital 3D shapes, work on how to extract, store and handle semantic content about 3D models is still at the beginning. Undoubtedly, the development of tools for the management of knowledge related to 3D shapes is fundamental to foster the development of totally new approaches to 3D content creation, retrieval and usage. In this paper we describe AIM@SHAPE's view of the digital shape lifecycle, and explain novel approaches to tackle the semantic annotation of 3D shapes.

2 Introduction

In the last decade, the concept of multimedia has evolved from single-content type, mainly related to non-textual data (e.g., images, videos or audio), to truly multimedia content, which integrates multiple medium types. Research on multimedia, however, is largely devoted to content whose digital representation is at most two-dimensional (e.g., images), possibly with the addition of time and audio (e.g., videos). At the same time, Computer Graphics has reached quite a mature stage where fundamental problems related to the modelling, manipulation and visualization of static and dynamic 3D shapes are well understood and solved. Considering that most PCs connected to the Internet are now equipped with high-performance 3D graphics hardware, it seems clear that in the near future 3D data will represent a huge amount of traffic and data stored and transmitted using Internet technologies. It has been predicted that *geometry is poised to become the fourth wave of digital-multimedia communication,* where the first three waves were sounds in the 70's, images in the 80's, and videos in the 90's.

There is a phenomenal activity around digital shapes. Currently in widespread use for computer-aided design and manufacture, they are becoming crucial to genomic, proteomic and medical modelling. In areas like culture, education and entertainment, shapes are equally essential in developing convincing virtual worlds. They are used extensively to develop models and create simulations, and to devise new designs that conform to engineering constraints, yet remaining functional and aesthetically pleasing. In personal entertainment, it is possible already today to buy your own 3D character for some computer games.

If most of the efforts in multimedia are currently focused on solving problems related to image-oriented content, the next step is to add a new dimension, i.e. 3D or time varying 3D, to this content and endow it with semantics. The impact of 3D content is comparable to the one of images, with a number of distinctive properties. 3D shapes offer more potential for interactivity since they can be observed and manipulated from different viewpoints. Also, the richness of their representation potentially contains more knowledge about an object than a simple picture. At the same time, representing a complex shape is known to be highly non trivial, due to the sheer mass of information involved and the complexity of the knowledge a shape can reveal. Therefore, we need tools for making digital shapes machine-understandable and not just human-understandable as today, developing semantic mark-up of content, intelligent agents and ontology infrastructures for fully-3D content.

The description of a shape is intrinsically not unique and varies according both to the application and user contexts. Therefore, the abstraction levels used to process or reason about 3D media should correspond to the mental models used to answer questions such as "what does it look like?", "what is its function?", thus making it possible to model, manipulate and compare the various 3D shapes in a semantics-oriented framework.

According to our experience, applications dealing with 3D shapes need a description of the media content and semantics in terms of knowledge related at least to the following types, or forms:

- *knowledge related to the geometry of 3D shapes*: while the descriptions of a digital 3D media can vary according to the contexts, the geometry of the object remains the same and it is captured by a set of geometric and topological data that define the digital shape;

- *knowledge related to the application domain in which 3D shapes are manipulated*: the application domain casts its rules on the way the 3D shape should be represented, processed, and interpreted; features are the key entities to describe the media content, and these are obviously dependent on the domain. Beside the description of the shapes content, a big role is played by knowledge of the domain experts which is used to manipulate the digital model: for example, the correct manner to compute a finite element mesh of a 3D object represented by free-form surfaces is subject also to informal rules that should be captured in a knowledge formalisation framework;

- *knowledge related to the meaning of the object represented by the 3D shapes*: they may represent objects that belong to a category of shapes, either in broad unrestricted domains (e.g. chair, table in the house furniture) or narrow specific domains (e.g. T-slots, pockets in mechanical engineering).

The first bullet is concerned with knowledge, which has geometry as its background domain. 3D geometry, as used in applications, has to do with a much richer variety of methods and models, and for example in the product modelling scenario, users might have to deal with different representation schemes for the same product within the same modelling pipeline.

The second bullet refers to knowledge pertaining to the specific application domain, but it has to be linked to the geometric content of the 3D shapes. Therefore, if we want to devise semantic 3D systems, with some reasoning capabilities, we have to formalise also expert knowledge owned by the professionals of the field.

Finally, the third bullet has to do with knowledge related to the existence of categories of shapes; as such, it is related both to generic and specific domains. Usually in 3D applications, it is neither necessary nor feasible to formalise the rules that precisely define these categories in terms of geometric properties of the shape, besides very simple cases. However, due to the potential impact of methods for search and retrieval of digital 3D models, there is a growing interest in methods that can be used to derive feature vectors or more structured descriptors that could be used to automatically classify 3D shapes.

3 The AIM@SHAPE Vision and Mission

In this context, the FP6-IST Network of Excellence AIM@SHAPE [1] is pursuing the introduction of knowledge management techniques in shape modelling, with the aim of making explicit and sharable the knowledge embedded in multi-dimensional media, with focus on 3D content. On the one hand, this requires the development of tools able to extract semantics from 3D models (e.g. automatic or semi-automatic annotation tools), on the other hand it is necessary to build a common framework for reasoning, searching and interacting with the semantic content related to the knowledge domain. One of the objectives of AIM@SHAPE is therefore to develop new methods and tools for modelling, extracting and reasoning about knowledge related to digital 3D content, where knowledge is concerned with *geometry* (the spatial extent of objects), *structure* (object features and part-whole decomposition), *attributes* (colours, textures, names attached to an object, its parts and/or its features), *semantics* (meaning or purpose in a specific context), and has interaction with *time* (e.g., shape morphing, animation, videos).

An example of semantics-based shape modelling is illustrated in Figure 1, where a bottom-up pipeline for modelling a virtual human is shown. The modelling process starts with the scanning of a real body model (a), and the acquired data are used to build a first digital model of the real shape (b). The geometry of the body is represented in this case by a triangle mesh, which contains all data needed to render nicely the digital object. In the triangle mesh, however, nothing is stored about the semantics of the objects or of its features: it is not possible to distinguish points belonging to the legs from points belonging to the hands. With suitable shape analysis methods, it is possible to detect relevant parts of the digital model, having a protrusion-like form (c); based on this analysis the initial geometry is segmented and the triangles are organized in a skeleton-like structure of the body model (d). Finally, another step of analysis is used, which makes use of context-specific rules, to tag parts of the structure with semantically-oriented labels, such as legs, arms and so on. The tagged model is now ready for being animated properly in a virtual environment scenario (e).

The shift from a purely geometric to a semantic-aware level of 3D content production and storage requires fundamental research to be done within an underlying common conceptualisation framework, which formalizes shape knowledge via the adoption of shared metadata and ontologies. In AIM@SHAPE, ontologies are structured frameworks of concepts, meanings and relations which make explicit the knowledge associated with shapes (for example, see Figure 2). They predefine semantics that can be used to annotate shapes with domain-specific information.

4 A new proposal for part-based annotation of 3D shapes

Annotating shapes amounts to formally coding additional knowledge in the form of structured attributes, or metadata, and is a crucial ingredient for the development of effective search mechanisms. Within future knowledge bases of annotated 3d media, one could be able to answer queries of the type “find a shape containing two arms and two legs”, or even to ask “find legs” and obtain as results proper subparts of whole shapes. In principle, an expert in a particular domain should be able to identify significant features and to assign them a specific meaning. As an example, an engineer should be able to look at a surface mesh representing a scanned engine and identify which parts have a specific mechanical functionality.

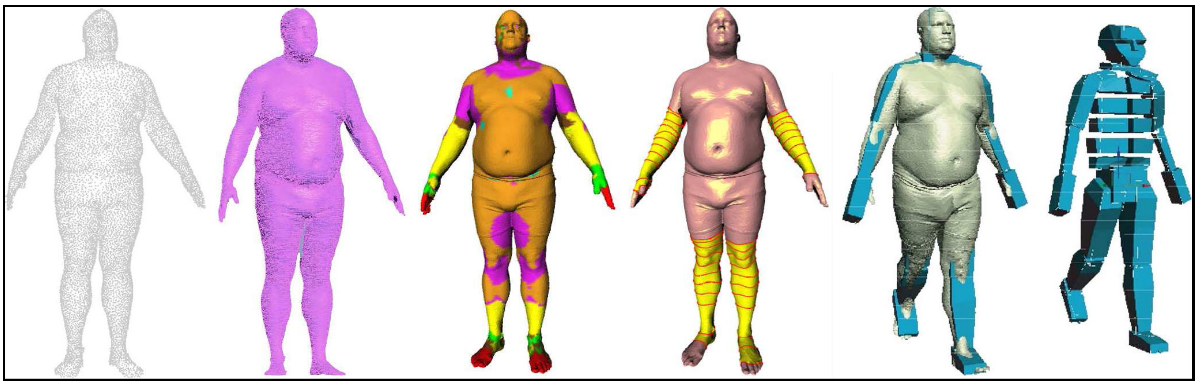

(a) (b) (c) (d) (e)

Figure 1: The bottom-up digital shape lifecycle applied to virtual humans.

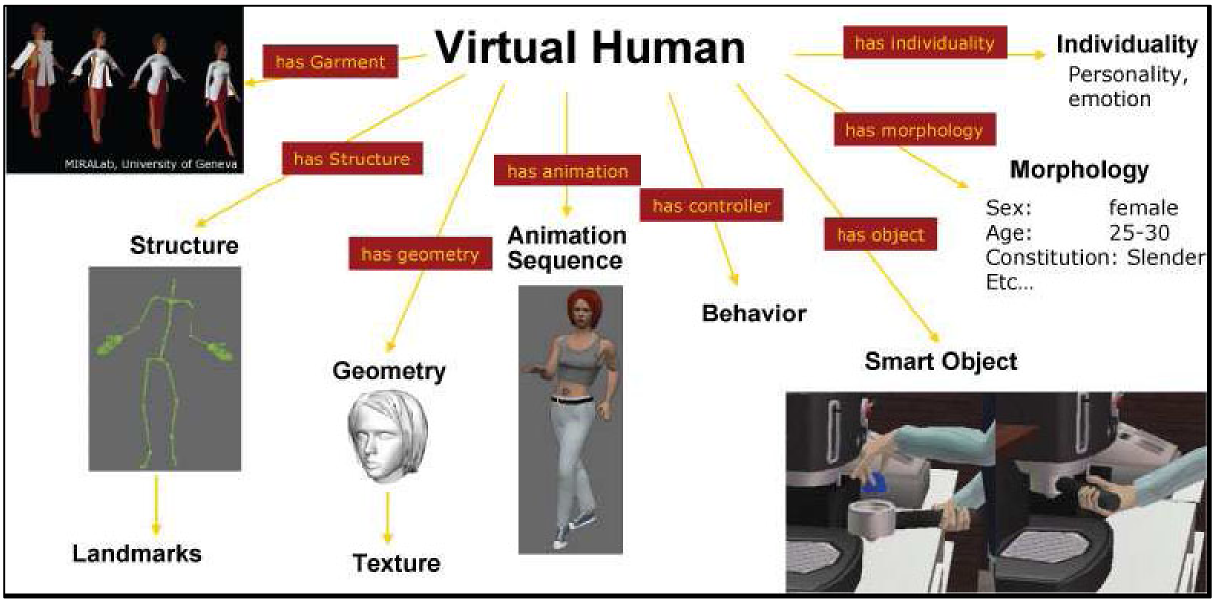


Figure 2: General scheme of the Virtual Human Ontology

Unfortunately, to the best of our knowledge, today there is no practical way to transform such expertise into usable content to be coupled with the plain geometric information. To bridge this gap, we defined an annotation pipeline and developed a prototype graphical tool called the *ShapeAnnotator*. 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.

Broadly speaking, expressing the semantics of 3D shapes requires the identification of significant parts, or *features*, the specification of the domain of expertise using some kind of formalism, and the storage of the geometry plus its semantic description in a way that could be accessed easily, by humans as well as by software agents.

Clearly, the optimal solution would be the automatic annotation of 3D shapes, but this is generally unfeasible due the intrinsic difficulty of devising a system that automatically extracts all the relevant features of an object. In the area of shape segmentation, indeed, some significant results have already been obtained [2], but the problem is still open and remains of alive interest. In contrast, to the best of our knowledge, very few work has been done to automatically annotate shapes, and only for very specialized tasks [3][4]. In section 6 a method to automatically segment human body shapes is described.

5 The ShapeAnnotator

The paradigm behind the ShapeAnnotator is based on the twofold assumption that an effective annotation must be relative to a specific domain, and the definition of relevant shape feature must be relative to the same domain.

Thus, the input of the ShapeAnnotator is constituted of (1) the shape to be annotated, (2) an ontology representing the annotation domain, and (3) an optional set of shape segmentation *plug-ins* to extract specific, domain-dependent features. Plug-ins are optional because the ShapeAnnotator already includes several standard segmentation algorithms whose results can be composed and edited within a *multi-segmentation* to define non-trivial features. In many cases, in fact, shape features are not sharply defined in terms of their boundary; if the annotation domain describes a head and a torso to be adjacent parts of a human body, for example, the neck should be considered part of both. In general, features may overlap and they do not necessarily form a partitioning of the whole (i.e. some parts may remain undefined as they do not have any particular meaning in the context addressed).

Note that for 2D images, segmentation algorithms are not always considered helpful to define features for annotation; on a flat image, in fact, useful features may be even sketched by hand [5]. In contrast, a 3D shape may be very complex and drawing the boundary of a feature might become a rather tedious task, involving not only the drawing stage, but also rotating the scene, translating and zooming in and out to show the portions of the surface to draw on.

Once segmentation algorithms have been run to properly define interesting features, the next step consists of selecting each such feature and tagging it with a concept of the ontology. To this aim, the ShapeAnnotator provides an integrated ontology browser to seek and select the proper concept by navigating across ontological relations, and to create an instance for the resulting **knowledge base**.

The output of the annotation process is a pair of files that encode:

- The geometry of the shape and its interesting features;
- The set of instances describing the features.

The domain ontology coupled with the instances form a knowledge base in which the semantics is connected to the geometry (see Figure 3).

6 Automatic annotation of human body models

As mentioned in section 4, at present automatic semantic annotation is not reachable in the general case, while it is possible in a few specific domains. In the following, we are going to describe an algorithm to automatically annotate parts of human body models and show the results on real scan data.

Basic components of articulated shapes such as human body models are best identified by tubular and non-tubular features. In fact, while geometric attributes may vary from a model to another, the human body structure is well defined and the basic components are predominantly tubular (e.g., arms, legs, fingers, neck). A segmentation into tubular/ non-tubular parts may be expressive enough to allow an automatic annotation of components with semantic content, at least in well specified knowledge domains like that of human body models.

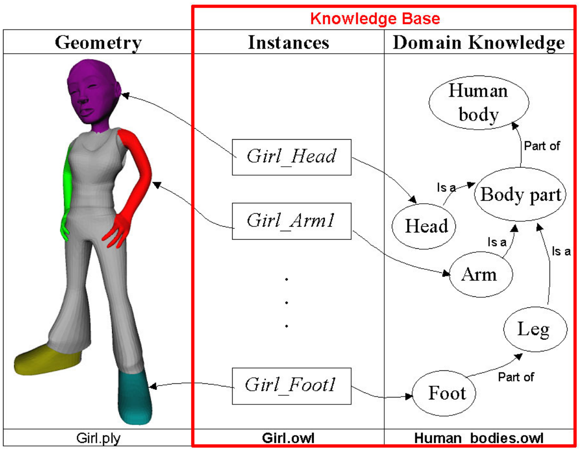


Figure 3: Bridging geometry and semantics. The file *girl.owl* encodes instances of concepts formalized in *Human_bodies.owl* and, at the same time, points to the corresponding geometry in *girl.ply*.

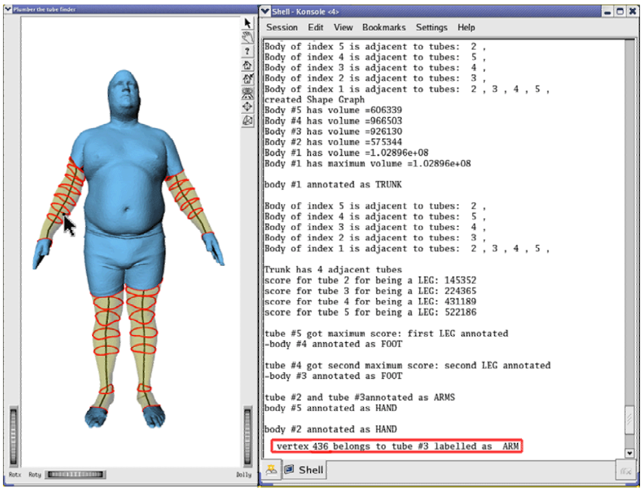
Therefore, we implemented an automatic semantic annotator for human body parts based on the segmentation given by an algorithm called *Plumber* [6]. Plumber segments a mesh into *tubes* and *blobs* (non-tubular features). For each features some geometric attributes are also computed, such as blobs' volume and tubes' axis length and section size.

The annotation can be defined as a function $f: S \rightarrow L$ from the set of segments S into the set of labels L. In our case, the segments are those given by *Plumber*; labels are defined in order to make the annotation exhaustive with respect to the segmentation, therefore:

L := *trunk, arm, hand, palm, finger, fingertip, leg, foot, neck, head.*

In general, some of the labels in S might not appear in the annotation because they have not been identified by the segmentation due to the posture, the poor quality of the scans, or the selection of level of details which do not enable to characterize small features such as fingers. In this last case, the hand segment will be labelled as *hand*, discarding the *palm*, *fingers*, and *fingertip* labels. Conversely, fingers, fingertips, and palm will be instantiated at the expense of *hand*, unless we deduce afterwards that adjacent regions labelled as *palm*, *finger*, and *fingertip* form a *hand*. The annotator exploits the geometric attributes of parts, computed during the segmentation phase. For tubes these are the axis length and the maximum, minimum and average length of cross sections, while for blobs the volume is considered. We point out that a tube segment has always two adjacent segments, while a blob segment may be adjacent to one or more parts; in particular, we will call *cap* a blob segment adjacent to one segment exactly. Given a segmented shape, we define as *shape-graph* the graph whose nodes are the identified patches and the arcs code the adjacency among them. The a-priory knowledge on human anatomy is exploited to define annotation rules of parts, based on geometric attributes and mutual adjacency relations of segments. The annotation rules come from the following considerations and imply a sequence of applications:

- the *trunk* is the blob segment of maximum volume[1];
- if the *trunk* is adjacent to four tubes, those are *legs* and *arms*; if it is adjacent to five tubes, also the *neck* has been segmented;
- if the *neck* has been segmented, it is the tube adjacent to the *trunk*, also adjacent to a cap, having minimum length; the *head* is the cap adjacent to the *neck*;
- among the four tubes adjacent to the *trunk*, not yet labelled (i.e. except the *neck*, if segmented) *arms* are those having maximum length, maximum section (greater value of maximum section length) and adjacent to a cap, that will be labelled as *foot*.
- the two tubes adjacent to the *trunk* still unlabeled will be annotated as *arms*;
- if a cap is adjacent to an *arm*, it will be labelled as *hand*; otherwise, the body segment adjacent to an *arm* (beyond the *trunk*) will be annotated as *palm*, and its adjacent tubes as *fingers*.
- Finally, caps adjacent to *fingers* will be annotated as *fingertips*.



```
Body of index 5 is adjacent to tubes:  2 ,
Body of index 4 is adjacent to tubes:  5 ,
Body of index 3 is adjacent to tubes:  4 ,
Body of index 2 is adjacent to tubes:  3 ,
Body of index 1 is adjacent to tubes:  2 , 3 , 4 , 5 ,
created Shape Graph
Body #5 has volume =606339
Body #4 has volume =966503
Body #3 has volume =926130
Body #2 has volume =575344
Body #1 has volume =1.02896e+08
Body #1 has maximum volume =1.02896e+08

body #1 annotated as TRUNK

Body of index 5 is adjacent to tubes:  2 ,
Body of index 4 is adjacent to tubes:  5 ,
Body of index 3 is adjacent to tubes:  4 ,
Body of index 2 is adjacent to tubes:  3 ,
Body of index 1 is adjacent to tubes:  2 , 3 , 4 , 5 ,

Trunk has 4 adjacent tubes
score for tube 2 for being a LEG: 145352
score for tube 3 for being a LEG: 224365
score for tube 4 for being a LEG: 431189
score for tube 5 for being a LEG: 522186

tube #5 got maximum score: first LEG annotated
-body #4 annotated as FOOT

tube #4 got second maximum score: second LEG annotated
-body #3 annotated as FOOT

tube #2 and tube #3annotated as ARMS
body #5 annotated as HAND

body #2 annotated as HAND
  vertex 436 belongs to tube #3 labelled as  ARM
```

Figure 4: automatic annotation of a human body scan. Selecting a point on a segment makes the corresponding label (“arm” in the example) to be printed on the screen.

Once the model is annotated, a mouse click over a segment will cause the corresponding label to be printed on the screen. In Figure 4, the graphical user interface of the whole work-flow (morphological analysis, segmentation, and annotation) is shown, side by side with the command shell where the main computation steps are reported by the program; also, the output of some queries on segment labelling by the user are displayed.

7 REFERENCES

[1] Network of Excellence: AIM@SHAPE - Advanced and Innovative Models And Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects. EC FP6 IST NoE Contract n. 506766. Duration: 2004-2007. http://www.aimatshape.net

[2] M. Attene, S. Katz, M. Mortara, G. Patané, M. Spagnuolo, A. Tal: Mesh segmentation-a comparative study. In SMI ’06: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006 (SMI’06), IEEE Computer Society.

[3] K.H. Hohne, A. Pommert, M. Riemer, T. Schiemann, R. Schubert, U. Tiede, W. Lierse: Anatomical atlases based on volume visualization. In VIS ’92: Proceedings of the 3rd conference on Visualization’92 (Los Alamitos, CA, USA, 1992), IEEE Computer Society Press, pp. 115–122.

[4] M. Kallmann, D. Thalman: Modeling objects for interaction tasks. In EGCAS’98: Proceedings of the 9th Eurographics Workshop on Computer Animation and Simulation (1998), pp. 73–86.

[5] K. Petridis, D. Anastasopulos, C. Saathoff, N. Timmermann, I. Kompatsiaris, S. Staab: M-OntoMat-Annotizer: Image annotation. linking ontologies and multimedia low-level features. In KES 2006 - 10th Intnl. Conf. on Knowledge Based, Intelligent Information and Engineering Systems (oct 2006).

[6] M. Mortara, G. Patanè: Plumber: a multi-scale decomposition of 3D shapes into tubular primitives and bodies. Proceedings of *ACM Symposium on Solid Modeling and Applications* SM'04 (Genova, Italy, 9-11 June 2004), pp. 339-344. G. Elber, N. Patrikalakis, P. Brunet (eds.). Eurographics Association, 2004.

---

[1] This statement always holds: if the model is undersegmented, the trunk segment will have the maximum volume, at the expense of the arms, legs, and neck.