3D Searching

Sree Lakshmi Addepalli
Department of Computer Engineering
Vivekanand Education Society's Institute Of Technology
Chembur, Mumbai
lakshi.ana@gmail.com

Sree Gowri Addepalli
Department of Computer Engineering
Vivekanand Education Society's Institute Of Technology
Chembur, Mumbai
gauriaddepalli@gmail.com

***Abstract*— from computer-aided design (CAD) drawings of complex engineering parts to digital representations of proteins and complex molecules, an increasing amount of 3D information is making its way onto the Web and into corporate databases. Because of this, users need ways to store, index, and search this information. Typical Web-searching approaches, such as Google's, can't do this. Even for 2D images, they generally search only the textual parts of a file, editor of the online Search Engine Showdown newsletter. However, researchers at universities such as Purdue and Princeton have begun developing search engines that can mine catalogues of 3D objects, such as airplane parts, by looking for physical, not textual, attributes. Users formulate a query by using a drawing application to sketch what they are looking for or by selecting a similar object from a catalog of images. The search engine then finds the items they want. The 3D-search system uses algorithms to convert the selected or drawn image-based query into a mathematical model that describes the features of the object being sought. This converts drawings and objects into a form that computers can work with. The search system then compares the mathematical description of the drawn or selected object to those of 3D objects stored in a database, looking for similarities in the described features. 3D search engine helps to have an idea of 3D model of anything on web. Hence, within many research and development areas like defence, navy, aero modelling, graphics and movie industries 3D search could prove useful and successful.**

Keywords—CAD; Search Engine; 3D objects; Image based Query; Algorithms

I. Introduction

Over the last few decades, computer science has made incredible progress in computer aided retrieval and analysis of multimedia data. For example, suppose you want to obtain an image of a horse for a Power point presentation. A decade ago, you could: 1) draw a picture, 2) go to a library and copy a picture, or 3) go to a farm and photograph a horse. Today, you can simply pick a suitable image from the millions available on the web. Although web search is commonplace for text, images, and audio, the information revolution for 3D data is still in its infancy. However, three recent trends are combining to accelerate the proliferation of 3D models, leading to a time in the future when 3D models will be as ubiquitous as other multimedia data are today: (1) new scanners and interactive tools are making construction of detailed 3D models practical and cost effective, (2) inexpensive graphics hardware is becoming faster, causing an increasing demand for 3D models from a wide range of people, and (3) the web is facilitating distribution of 3D models.

A. Need for 3D Search Engine

Now days, developments are changing the way we think about 3D data. For years, a primary challenge in computer graphics has been how to construct interesting 3D models. In the near future, the key question will shift from "how do we construct them?" to "how do we find them?" For example, consider a person who wants to build a 3D virtual world representing a city scene. He will need 3D models of cars, street lamps, stop signs, etc. Will he buy a 3D modeling tool and build them himself? Or, will he acquire them from a large repository of 3D models on the Web? We believe that research in retrieval, matching, recognition, and classification of 3D models will follow the same trends that can already be observed for text, images, audio, and other media. An important question then is how people will search for 3D models. Of course, the simplest approach is to search for keywords in filenames, captions, or context. However, this approach can fail: (1) when objects are not annotated (e.g., "B19745.wrl"), (2) when objects are annotated with in specific or derivative keywords (e.g. "yellow.wrl" or "sarah.wrl"), (3) when all related keywords are so common that the query result contains a flood of irrelevant matches (e.g., searching for "faces"– i.e., human not polygonal), (4) when relevant keywords are unknown to the user (e.g., objects with misspelled or foreign labels), or (5) when keywords of interest were not known at the time the object was annotated. In these cases and others, a 3D search engine is needed.

B. How to Search for 3d Models

We hypothesize that shape-based queries will be helpful for finding 3D objects. For instance, shape can combine with function to define classes of objects (e.g., round coffee tables). Shape can also be used to discriminate between similar objects (e.g., desk chairs versus lounge chairs). There are even instances where a class is defined entirely by its shape (e.g., things that roll). In these instances, "a picture is worth a thousand words."

The challenges are two-fold. First, we must develop computational representations of 3D shape (shape descriptors) for which indices can be built and similarity queries can be answered efficiently. In this paper, we investigate combinations of 3D sketching, 2D sketching, text, and interactive refinement based on shape similarity.

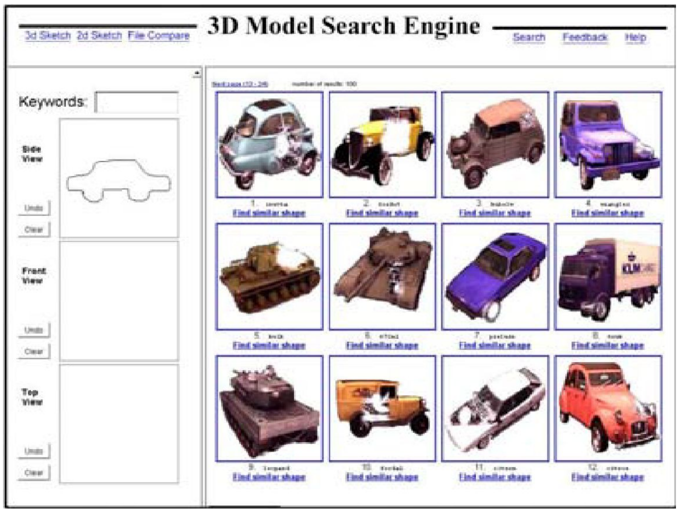

Figure 1: A Model 3D Search Engine

It allows a user to specify a query using any combination of keywords and sketches (left). Then, for each query, it returns a ranked set of thumbnail images representing the 16 best matching 3D models.

II. System overview for 3d models

Execution proceeds in four steps: crawling, indexing, querying, and matching. The first two steps are performed off-line, while the last two are done for each user query. The following text provides an overview of each step and highlights its main features:

1) Crawling: a focused crawler that incorporates a measure of 3D model "quality" into its page rank. Using this crawler, We augment this database with 2,873 commercial models provided by 3D vendors.

2) Indexing: We compute indices to retrieve 3D models efficiently based on text and shape queries.

3) Querying: We allow a user to search interactively for 3D models. Our system supports query methods based on text keywords, 2D sketching, 3D sketching, model matching, and iterative refinement.

4) **Matching:** For each user query, the web server uses its index to return the sixteen 3D models that best match the query.

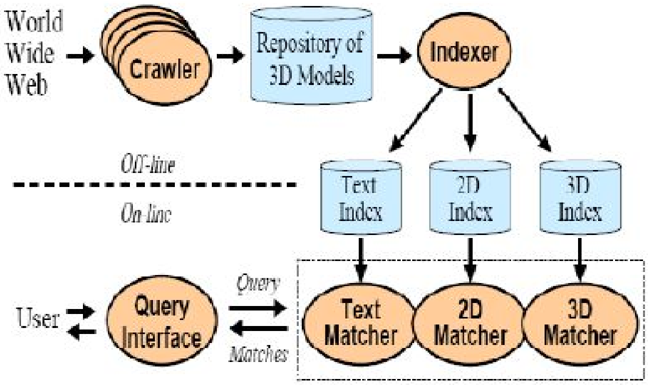

Figure 2: System Organization

C. Sketch Query

*S*hape similarity queries are only possible when *the user* already has a representative 3D model. *In some* cases, he will be able to fi*nd one by* using a text search. However, in other cases, he will have to create it from scratch. An interesting open question then is "What type of modeling tool should be used to create shapes for 3D retrieval queries?*"* Rather than providing a tool with which a trained user can create models with exquisite detail and/or smoothness properties, our goal is to allow novice users to specify coarse 3D shapes quickly. In particular, the interface should be easy to learn for first time visitors to a website. Of course, this requirement rules out almost every 3D modelling tool available today–i.e., it would not be practical to require everybody who wants to use a 3D search engine to take a three week training course to learn the complicated menu structure of a commercial CAD tool. Instead, we have investigated two alternatives.

Figure3: 3D Sketch Query Interface

The first approach is to specify shape queries with a simple 3D sketching tool, such as Teddy or Sketch. To investigate this approach, a query interface is developed in which the user creates a simple 3D model with Teddy, and then the system retrieves similar models. Unfortunately, experiences suggest that even its simple gesture interface is still too hard for novice and casual users to learn quickly. Moreover, only certain types of shapes can be created with Teddy. Hence making 3D tools even simpler would require further constraints on the types of shapes that could be produced. Thus, motivation to look for alternate sketching paradigms is taking place.

The second approach is to draw 2D shapes with a pixel paint program and then have the system match the resulting image(s) to 2D projections of 3D objects. The main advantage of this approach is that the interface is easy to learn. All but the most novice computer users have used a 2D paint program before, and there are no complicated viewing or manipulation commands. Of course, the main disadvantage is that 2D images generally have less shape information than 3D models. Hence compensation for this factor somewhat by allowing the user to draw multiple 2D projections of an object in order to better define its shape.

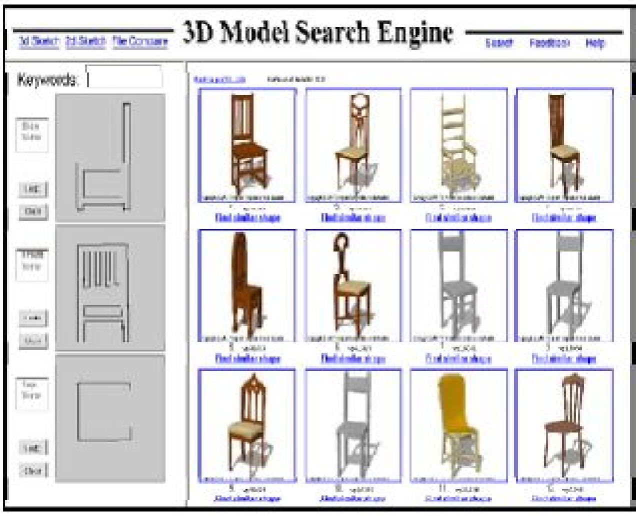

Figure 4: 2D Sketch Query Interface

B. Text Query

This system also supports searching for 3D models by matching keywords in their textual descriptions. To support this feature, we construct a representative document for each 3D model. The text in that document includes the model filename, the anchor and nearby text parsed from its referring Web page, and ASCII labels parsed from inside the model file.

Figure 5: Text Query Interface

C. Multi-Model Query

Since text and shape queries can provide orthogonal notions of similarity corresponding to function and form, this search engine allows them to be combined. We support this feature in two ways. First, text keywords and 2D/3D sketches maybe entered in a single multimodal query. Second, text and shape information entered in successive queries can be combined so that a user can refine search terms adaptively. For instance, if a user entered text keywords in a first query, and then clicked a “Find Similar Shape” link, the text and 3D shape would combine to form a second query. These types of multimodal queries are often helpful to focus a search on a specific subclass of objects. For example, a query with only keywords can retrieve a class of objects (e.g., tables),but it is often hard to home in on a specific subclass with text alone (e.g., round tables with a single pedestal). Similarly, a query with only a sketch can retrieve objects with a particular shape, but it may include objects with different functions (e.g., both tables and chairs).Multimodal input can combine ways of describing objects to form more specific queries.

Figure 6: Multi-Model Query Interface

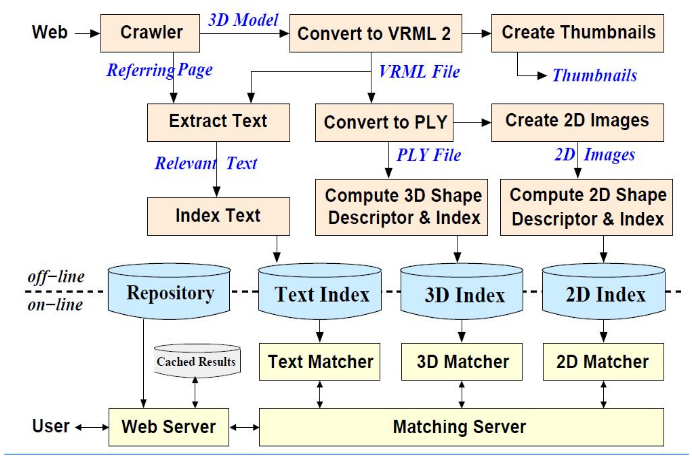

Figure 7: Flow of data through the Search Engine

III. FUTURE WORK

This paper has just scratched the surface of research on shape-based retrieval and analysis for computer graphics. The following are just a few of the many topics that deserve further investigation:

- **New query interfaces:** it will be interesting to consider other methods for specifying shape based queries. For instance, the following constraint-based description might be used to retrieve 3D models of a chair: "give me objects consisting of a box-shaped seat with four equal length and nearly cylindrical legs attached to the bottom side of each corner and a box-shaped back above the seat with width matching that of the seat, etc." This approach captures parameterized classes of objects with a compact description.
- **New matching and indexing algorithms:** follow-up work should consider other types of shape matching problems. For instance, we currently compare whole objects, but it would be interesting to match partial objects as well. They could be used to find a car within a city scene or to find a Mercedes by looking for its hood ornament. Other matching algorithms might consider attributes of 3D models, including colour, texture, structure, and animations.
- **New modelling tools:** future 3D modelling systems should consider integrating shape-based matching and retrieval methods into interactive sketching tools. For instance, consider a 3D model synthesis paradigm in which a user draws a rough sketch of a desired 3D model and the system "fills in the details" semi-automatically by suggesting matching detailed parts retrieved from a large database. In such a paradigm, the user could retain much of the creative control over model synthesis, while the system performs most of the tedious tasks required for providing model detail.
- **New applications:** it would be interesting to see whether the shape-based query and indexing methods described in this paper can be used for other applications, such as in mechanical CAD, medicine, and molecular biology.

In the near future, we expect that shape-based retrieval and analysis of 3Dmodels will become a very important research area in computer graphics. This paper makes a small step in that direction.

IV. LIMITATION

Better 2D image matching methods: 2D sketching interface would be more effective with better image matching algorithms. Sometimes users create query sketches with interior texture and/or details which are "interpreted" wrongly and unexpected results are sometimes returned to the user. Of course, this problem could be rectified somewhat by providing users with instructions or examples about how to draw their sketches.

New modelling tools: future 3D modelling systems should consider integrating shape based matching and retrieval methods into interactive sketching tools. For instance, consider a 3D model synthesis paradigm in which a user draws a rough sketch of a desired 3D model and the system "fills in the details" semi-automatically by suggesting matching detailed parts retrieved from a large database. In such a paradigm, the user could retain much of the creative control over model synthesis, while the system performs most of the tedious tasks required for providing model detail.

V. CONCLUSION

In summary, it investigates issues in building a search engine for 3D models. The main research contributions are:(1) New query interfaces that integrate text, 2D sketches, 3D sketches, and 3D models. (2) We provide a large repository of 3D models and a way to find the interesting ones.

REFERENCES

[1] Thomas Funkhouser, Patrick min, Michael Kazhdan,Joyce Chen, Alex Halderman, and David Dobkin Princeton University, "A Search Engine for 3D Models" ACM Transactions on Graphics, Vol. V, No. N, 10 202002,.

[2] Zeleznik, R. C., Herndon, K. P., and Hughes, J. F. 1996. Sketch, "An Interface for Sketching 3D Scenes", In Proceedings of SIGGRAPH 96. Computer Graphics Proceedings, Annual Conference Series.

[3] Igarashi, T.,Matsuoka, S., and Tanaka, H. 1999. Teddy: "A Sketching Interface for 3D Freeform Design". In Proceedings of SIGGRAPH 1999. Computer Graphics Proceedings, Annual Conference Series., Los Angeles, CA, 409–416.