Pers Ubiquit Comput (2009) 13:59–67
DOI 10.1007/s00779-007-0166-z

ORIGINAL ARTICLE

3D content-based search using sketches

Konstantinos Moustakas · Georgios Nikolakis · Dimitrios Tzovaras · Sebastien Carbini · Olivier Bernier · Jean Emmanuel Viallet

Received: 11 July 2006 / Accepted: 2 January 2007 / Published online: 20 April 2007

Abstract This paper presents a novel interactive framework for 3D content-based search and retrieval using as query model an object that is dynamically sketched by the user. In particular, two approaches are presented for generating the query model. The first approach uses 2D sketching and symbolic representation of the resulting gestures. The second utilizes non-linear least squares minimization to model the 3D point cloud that is generated by the 3D tracking of the user's hands, using superquadrics. In the context of the proposed framework, three interfaces were integrated to the sketch-based 3D search system including (a) an unobtrusive interface that utilizes pointing gesture recognition to allow the user manipulate objects in 3D, (b) a haptic–VR interface composed by 3D data gloves and a force feedback device, and (c) a simple air–mouse. These interfaces were tested and comparative results were extracted according to usability and efficiency criteria.

Keywords 3D content-based search · Multimodal interfaces · Sketch

K. Moustakas · G. Nikolakis · D. Tzovaras
Informatics and Telematics Instiutte, 1st Km Thermi-Panorama Road, 57001 Thermi-Thessaloniki, Greece
e-mail: gniko@iti.gr

D. Tzovaras
e-mail: tzovaras@iti.gr

K. Moustakas (✉)
Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, 54006 Thessaloniki, Greece
e-mail: moustak@iti.gr

S. Carbini · O. Bernier · J. E. Viallet
France Telecom R&D, Lannion 22307, France
e-mail: sebastien.carbini@rd.francetelecom.com

O. Bernier
e-mail: olivier.bernier@rd.francetelecom.com

J. E. Viallet
e-mail: jeanemmanuel.viallet@rd.francetelecom.com

1 Introduction

Search and retrieval (S&R) of 3D objects is nowadays a very challenging research topic and has application branches in numerous areas like recognition in computer vision and mechanical engineering, content-based search in e-commerce, and edutainment applications, etc. [1]. These application fields will expand in the near future, since the 3D model databases grow rapidly due to the improved scanning hardware and modeling software that have been recently developed.

The difficulties of expressing multimedia and especially 3D content via text-based descriptors, reduces the performance of the text-based search engines to retrieve the desired multimedia content. To resolve this problem, 3D content-based S&R has drawn a lot of attention in the recent years.

However, the visualization and processing of 3D models are much more complicated than those of simple multimedia data [2]. The major difference lies in the fact that 3D models can have arbitrary topologies and cannot be easily parameterized using a standard template, which is the case for images. Moreover, there can be many different models of representing them, i.e., indexed facets, voxel models, etc. Finally, processing 3D data is much more computationally intensive, than processing media of lower dimension, and often requires very large amounts of memory.

Many researchers worldwide are currently developing 3D model recognition schemes. A number of approaches exist in which 3D models are compared by means of measures of similarity of their 2D views [3]. More direct 3D model search methods focus on registration, recognition, and pairwise matching of surface meshes [4]. However, these methods require a computational costly search to find pairwise correspondences during matching. Significant work has also been done in matching 3D models using geometric characteristics, where initial configurations are derived from conceptual knowledge about the setup of the acquisition of the 3D scene [5] or found automatically by extracting features such as curvature or edges [6].

A typical S&R system, like the aforementioned ones, evaluates the similarities between query and target objects according to low-level geometric features. However, the requirement of a query model to search by example often reduces the applicability of an S&R platform, since in many cases the user knows what kind of object he wants to retrieve but does not have a 3D model to use as query.

Imagine the following use case: the user of a virtual assembly application is trying to assemble an engine of its spare parts. He inserts some rigid parts into the virtual scene and places them in the correct position. At one point he needs to find a piston and assemble it to the engine. In this case, he has to manually search in the database to find the piston. It would be faster and much easier if the user had the capability of sketching the outline of the piston using specific gestures combined with speech in order to perform the search. In the context of this project the integration of speech and gestures for the generation of the query model is addressed. Speech commands are used for performing specific actions, while gesture recognition is used to draw a sketch of the object and to manipulate the scene objects in the 3D space. The system is also capable to assemble the built objects so as to generate complex structures. The sketch-based 3D search engine has been tested using three interfaces. Comparative results on the usability and efficiency of the interfaces are presented in the experimental results section.

2 3D content-based search

For each 3D model, rotation invariant geometrical descriptors are extracted. In particular, the object is initially normalized in terms of translation and scaling, i.e., it is translated to the center of the coordinate system, and is scaled uniformly so that the coordinates of all its vertices lie in the interval [0,1]. Next, N concentric spheres are built centered at the origin of the coordinate system. Each sphere is built using tessellation of a normal icosahedron so that the vertices over its surface are uniformly distributed. In the experiments 20 concentric spheres of 16,002 vertices are used. For each sphere the discrete 3D signal $F(r_S,\ \theta_i,\ \varphi_i)$ is assumed, where i is the index of the sphere vertices. The values of function $F(r_S,\ \theta_i,\ \varphi_i)$ are calculated using the Spherical Trace Transform (STT) [7].

The extraction of the final descriptor vectors, which will be used for the matching algorithm, is achieved by applying the spherical functionals ''T,'' as described in [7], to the initial features $F(r_S,\ \theta_i,\ \varphi_i)$ generated from the STT. The spherical functionals for each concentric sphere ''ρ'' are summarized below:

$$T_1(F) = \max\{F(r_S, \theta_i, \phi_i)\}, \tag{1}$$

$$T_2(F) = \sum_{j=1}^{N_S} |F'(r_S, \theta_i, \phi_i)|, \tag{2}$$

$$T_3(F) = \sum_{j=1}^{N_S} F(r_S, \theta_i, \phi_i), \tag{3}$$

$$T_4(F) = \max\{F(r_S, \theta_i, \phi_i)\} - \min\{F(r_S, \theta_i, \phi_i)\}, \tag{4}$$

$$T_l(F) = A_l^2 = \sum_m a_{lm}, \tag{5}$$

where N_S is the total number of sampled points ($\eta_j, j = 1, ..., N_S$) at each concentric sphere, $l = 0, ..., L$ and $-l < m < l$. The values of a_{lm} are the expansion coefficients of the Spherical Fourier Transform [8]:

$$a_{lm} = \sum_{i=1}^{N_S} F(r_S, \theta_i, \phi_i) \times Y_{lm}(\eta_i)\frac{4\pi}{N_S}, \tag{6}$$

where Y_{lm} (η_i) corresponds to the spherical harmonic function, which is defined through:

$$Y_{lm}(\theta, \phi) = k_{l,m} P_l^m(\cos\theta)e^{jm\phi}, \tag{7}$$

where $P^m{}_l$ is the associated Legendre polynomial of degree l and order m, $k_{l,m}$ a normalization constant and j the imaginary unit.

The quantities $A^2{}_l$ are invariant to any rotation of the 3D model. Choosing a sufficiently large number of L coefficients of the Spherical Fourier Transform, a total number of $L + 4$ spherical functionals are used for each concentric sphere.

Finally, the descriptor vectors $D(l)$ are created, where $l = 0, ..., (L + 4)N_C$ is the total number of descriptors and N_C is the number of concentric spheres. In the experiments described in the sequel, $L = 26$ and $N_C = 20$ were chosen.

Now, let A, B, be two 3D part models and D_A, D_B, their descriptor vectors, respectively. The two parts are com-

pared in terms of similarity according to the following formula:

$$D_{\text{dissimilarity}} = \sqrt{\sum_{l=1}^{(L+4)N_C} |D_A(l) - D_B(l)|}. \quad (8)$$

Figure 1 depicts the retrieved objects using as input the first model of each column.

3 Sketching the query model

The sketch-based query model generation algorithm aims to provide the sufficient means for the easy and fast design of an approximation of the target model using primitive objects. The user can choose between two approaches, (a) the pseudo-3D sketching and (b) the interactive implicit 3D model sketching.

3.1 Pseudo-3D sketching

As also implied by the title, the pseudo-3D sketching approach is not a direct 3D sketching system. The user sketches initially a cross-section of a desired primitive (e.g., a circle when aiming at a cylinder) and then defines the size in the still undefined dimension. Pseudo-3D sketching consists of the following steps:

1. Sketching the 2D contour of the desired primitive object.
2. Choosing among the corresponding 3D shapes using speech commands [9] (e.g., for a circle choose between sphere, cylinder, and cone) and define its height, which cannot be drawn in 2D.
3. If a new primitive is desired go to Step 1, otherwise proceed to Step 4.
4. Assemble the primitives to form the final shape.

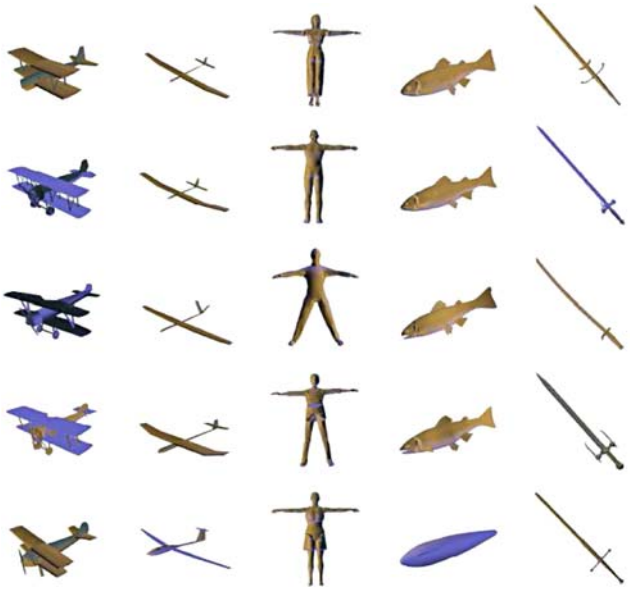

Fig. 1 3D search results using as query the models of the first row

The user initially sketches, using one of the sketching interfaces that will be described in the sequel, the 2D contour of the primitive to be inserted, e.g., circle for a sphere, a cylinder or a cone, rectangle for parallelepipeds, cubes, and triangles for a pyramid or a prisma. These shapes are recognized using least squares minimization with the Levenberg–Marquardt algorithm [10, 11] and a sample primitive is automatically inserted in the scene. Next, the degrees of freedom that cannot be defined just from the 2D sketch are defined and the primitive is manipulated. In other words the user defines the height of the object and translates, scales, rotates it until it reaches its target position. After inserting all the primitives they are assembled to the final target query model that is used as input to the 3D content-based search procedure described in Sect. 2. An example of the sketching procedure is illustrated in Fig. 2.

3.2 Interactive implicit 3D model sketching

The second tool provided to the user to sketch primitive 3D objects is a direct 3D sketching system. 3D points are extracted from the user's hand positions in consecutive frames. The resulting point clouds have in general the following characteristics:

- The point clouds are noisy, due to tracking noise when visually tracking the hand or due to the oscillation error when tracking the hands using magnetic trackers.
- When drawing in 3D, on contrary to 2D, it is not easy for the user to control his drawing. An additive error is observed that is in general increased at the last phases of the sketching procedure.

Due to these reasons it was decided that the system should illustrate the following characteristics:

- Robustness to noise: implicit surface approximation of the sketch is used so as to overcome the noise problem inherent in every real-time tracking procedure. Superquadrics were chosen as implicit surfaces due to the large number of shapes that they can efficiently transformed to.
- Direct control of the sketch: the system provides to the user direct control of the resulting sketch by displaying as feedback the best fit of the implicit surface at every moment of the sketching procedure.

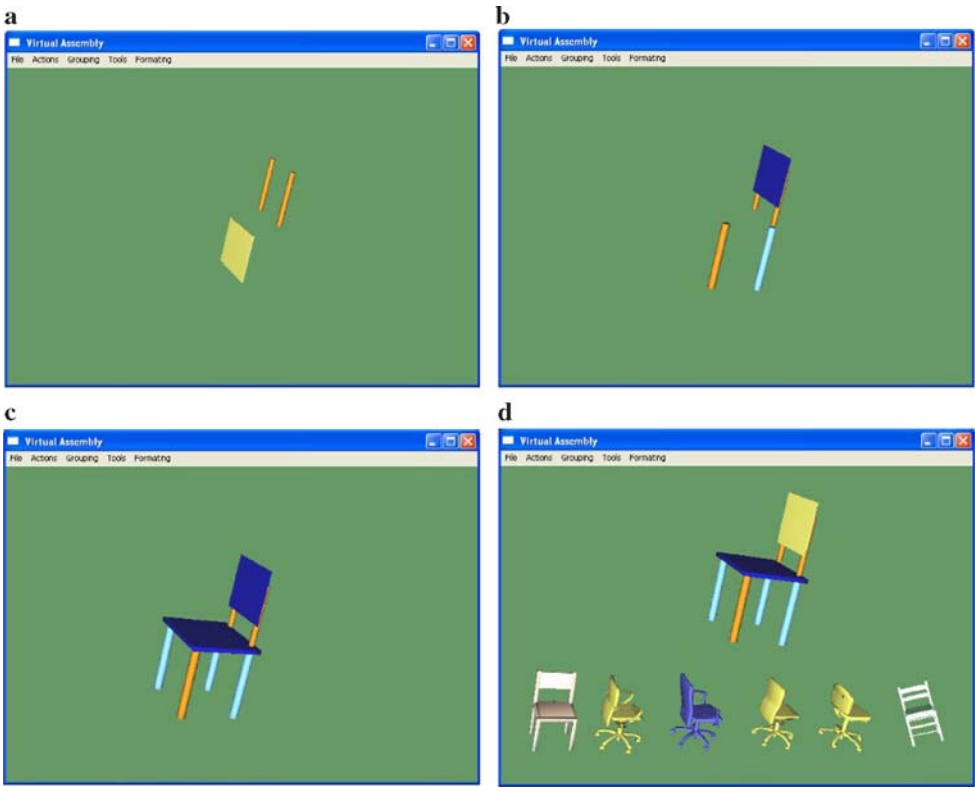

Fig. 2 **a–c** Sketching procedure and **d** 3D search results

It is obvious that it was decided to include a direct 3D sketching system instead of a symbolic one [12], that recognizes specific symbols that correspond to specific 3D promitives. The reason for this choice was to achieve complementarity with the pseudo-3D sketching approach that is actually a symbolic sketching method even if the shape characteristics that are sketched are included in the imported primitive.

3.2.1 Superquadric modeling

Superquadrics have been excessively used [13, 14] to model objects from range images and depth maps. Typically, they are a family of analytical implicit surfaces like superellipsoids, superparaboloids, superhyperboloids, and supertoroids. However, in the literature [13] the term superquadric is usually used to describe superellipsoids, due to their high applicability. Superquadrics (superellipsoids) are described by the following implicit equation.

$$F(x,y,z) = \left(\left(\frac{x}{\alpha_1}\right)^{2/\varepsilon_2} + \left(\frac{y}{\alpha_2}\right)^{2/\varepsilon_2}\right)^{\varepsilon_2/\varepsilon_1} + \left(\frac{z}{\alpha_3}\right)^{2/\varepsilon_1} = 1 \tag{9}$$

Function (9) is called inside–outside functions, because if for a 3D point (x, y, z) the evaluation of (9) yields $F(x, y, z) > 1$ the point lies outside the convex surface, if $F(x, y, z) < 1$ it lies inside and if $F(x, y, z) = 1$ it lies on the surface. Deformation parameters, which correspond to tapering, bending, etc. [13] can be added to the implicit equation so as to produce a more flexible model. Figure 3 illustrates four superquadrics for different values of ε_1 and ε_2.

After the selection of the appropriate superquadric equation to model the 3D data, the problem of modeling the 3D object using a superquadric reduces to the least squares minimization of the non-linear inside–outside function $F(x, y, z)$ with respect to several shape parameters. In particular,

$$F(x,y,z) = F\big(x,y,z;\alpha_1,\alpha_2,\alpha_3,\varepsilon_1, \varepsilon_2,\phi,\theta,\chi,t_x,t_y,t_z,K_x,K_y,k,\alpha\big), \tag{10}$$

where (x, y, z) is a point in the 3D space, α_1, α_2, α_3, ε_1, ε_2 are the superquadric shape parameters, φ, θ, χ and t_x, t_y, t_z are the Euler angles and translation vector coefficients, respectively, K_x and K_y are tapering deformation parameters and k, α the bending deformation parameters. The above parameters are determined so as to minimize the following mean square error.

$$\text{MSE} = \sum_{i=1}^{N} \sqrt{\alpha_1, \alpha_2, \alpha_3}(F(x_i, y_i, z_i) - 1)^2, \tag{11}$$

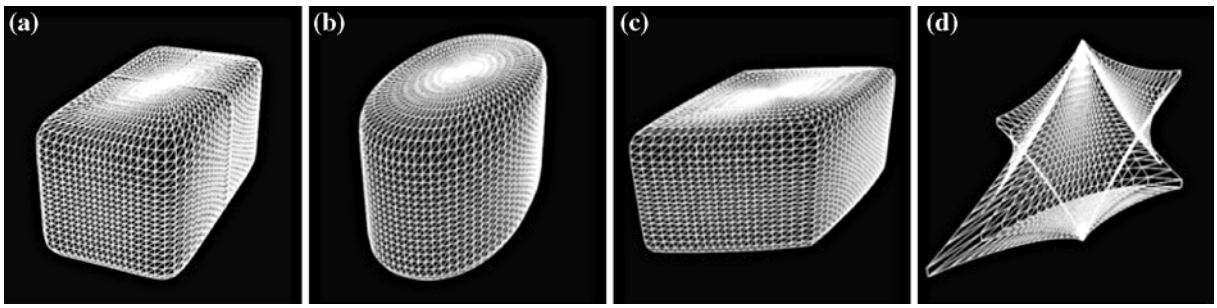

Fig. 3 Superquadrics with **a** $\varepsilon_1 = 0.2$, $\varepsilon_2 = 0.2$, **b** $\varepsilon_1 = 0.2$, $\varepsilon_2 = 0.9$, **c** $\varepsilon_1 = 0.2$, $\varepsilon_2 = 1.8$, and **d** $\varepsilon_1 = 2.4$, $\varepsilon_2 = 3.2$

where N is the number of the obtained 3D data. The term $\sqrt{\alpha_1, \alpha_2, \alpha_3}$ was introduced in [13] so as to accelerate the convergence of the minimization algorithm by transforming the parameter space to a steeper one.

The Levenberg–Marquardt method for non-linear least squares minimization, which is widely used [13, 14], is used in the present article in order to evaluate the shape parameters from the point clouds obtained at 3D sketching step.

3.2.2 Convergence-outlier removal

One big issue when using non-linear least squares minimization, especially when the parameter space is of high dimension like in the present case, is stability. When the data are noisy the minimization might not converge. In the context of the proposed framework, whenever instability is detected, the system enters the ''stabilization loop'' that removes all trajectory points that do not pass the following test:

$$|F(x, y, z) - 1| < \delta, \tag{12}$$

where δ is an experimentally selected threshold and the values of the superquadric parameters are the values of the last converging estimate. If after the test convergence can still not be achieved, δ is set to a lower value until the Levenberg–Marquardt method converges.

4 Query interfaces

The query interface to the 3D search engine is a multimodal gesture–speech interface. The following table describes, the actions that are controlled with gestures and with speech (Table 1).

4.1 Speech recognition

To recognize speech commands, the speech signal is linearly sampled at 8 kHz in 16 bits. Next, *Mel Frequency Cepstrum Coefficients* (MFCC) coefficients are computed, each 16 ms, on 32 ms signal frames. The recognition system uses the frame energy, eight cepstral coefficients and an estimation of the first- and second-order derivatives of the speech signal. Thus, the observation vector has 27 dimensions.

Table 1 Speech–gesture controlled actions

Speech controlled actions	Actions performed using gestures or automatically by the system
No speech	The 3D pointer follows the motion of the user's hand
Selection	Point at the object to be selected
Translation	Move the hand until the object reaches the target 3D position
Rotation	Rotate the hands like grabbing and rotating a sphere
Scaling	Increase decrease the distance between the hands
Sketching	Freehand sketching
Search	Use the selected object as query and search for similar content
Select group	Initiate grouping the primitives and call the selection command for each primitive
Retrieve	Retrieve the objects from the database starting with the most similar
Next	Retrieve next object
Delete	Delete selected object
Clone	Clone selected object
Stop action	Stop currently performed action

In the decoding system, Hidden Markov Models are used. The recognized sentences syntax is described in a grammar. The used vocabulary consists of 50 words. Dependent on the context, each word is obtained by phonetic units concatenation named allophones. The system finally outputs the n-best results [10].

The input signal is filtered using a noise/speech detector component to provide the decoder only with speech signal surrounded by silent frames. Beginning of detection is not causal. The detection component provides several frames, which precede the speech detection decision. But as the

decoder is faster than real time, it recovers from the non-causality of the detection component.

To detect end of speech, some consecutive silent frames must be observed. Thus, the best solution can be provided as soon as the last frame has been received. Computing the *n*-best solutions generate a negligible lag compared to the lag due to the silent frames. The number of frames to detect the end of speech is a parameter of the noise/speech component and is set to 15.

4.2 Unobtrusive gesture interface

The first interface is totally unobtrusive. The head and hands of the user are captured using a stereo camera and are efficiently tracked [10] using a statistical model composed of a color histogram and a 3D spatial Gaussian function [15], while the user sketches or performs specific actions.

Most people instinctively use the eye-tip of the finger line to point at a target. This convention is used in the present framework to estimate the pointing direction of the user. More precisely, the pointing area is estimated by the projection of the head–hand axis to the screen.

The first detected hand is tagged as ''pointing hand'' and the second as ''control hand.'' The system can be used by right-handed as well as by left-handed persons (predominant hand is generally used to point) without differentiating explicitly the right-hand from the left.

Once detected, body parts (head and hands) are tracked simultaneously until tracking failure. In this occasion, the algorithm automatically re-triggers detection for the lost part. The tracking process aims at representing each new image by a statistical model utilizing the EM algorithm [15]. The statistical model is composed of a color histogram and a 3D spatial Gaussian function for each tracked body part (Fig. 4).

During runtime, gesture recognition is triggered by speech commands. The axis, obtained from the first hand and the 3D position of the head, is used to compute the pointing direction. Once the pointing area on the screen is estimated, it can be used in a variety of actions. When the user utters ''selection'' the pointed 3D object is selected. Once selected, using the ''move'' command, the object keeps moving, following the hand's motion, while with the ''rotate'' command it rotates around its center of mass following the rotation of the axis defined by the 3D position of the two hands. Another example is the ''scale'' command. In this case the current 3D distance between the hands is taken as reference value and the current size of the object as reference size. Then, the object is continuously resized proportionally to the distance between the hands (i.e., when distance between hands is twice as large as the reference distance, the object is two times larger than its reference size). Finally, all actions are stopped when ''O.K.'' is uttered.

4.3 Virtual reality haptic interface

The backbone of this interface is a haptic glove that is used as input to the application, as it is capable of identifying hand gestures, and as output since it provides tactile or force feedback. It handles both human-hand movement input and haptic force-feedback for the fingers using Immersion's CyberGlove® (Fig. 5a) and CyberGrasp™ (Fig. 5b) haptic devices [16]. CyberGlove is a widely used human-hand motion-tracking device of proven quality. CyberGrasp is currently one of the very few force-feedback devices that are offered commercially, providing high quality of construction, operation and

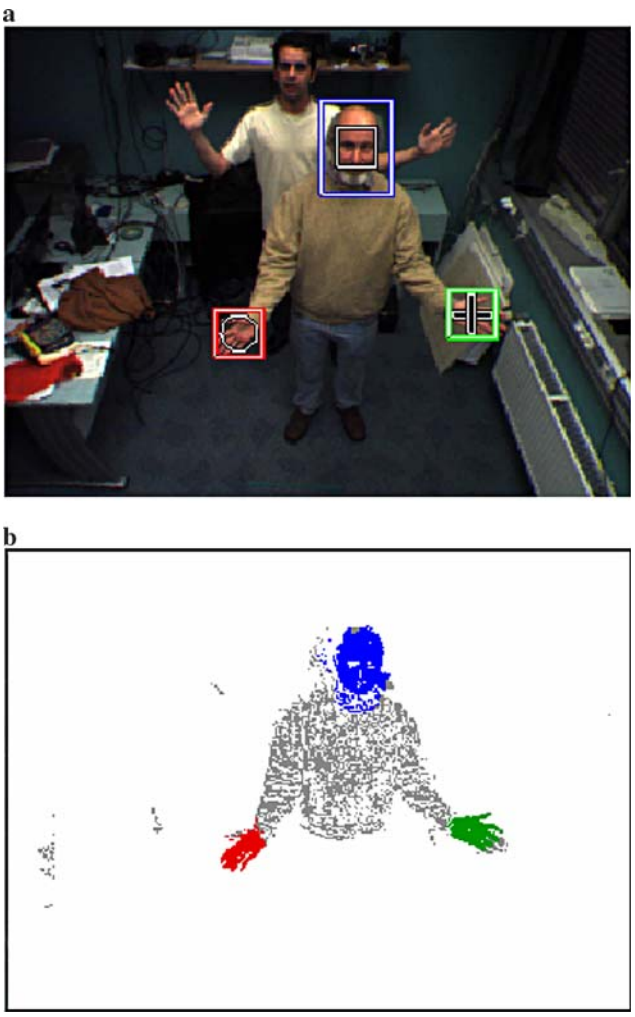

Fig. 4 **a** Camera image (*rectangle*: head, *circle*: hand1, and *cross*: hand2), **b** Observations assigned to one of the four models depending on their probabilities (*blue*: head, *red*: hand1, *green*: hand2, *gray*: discard, and *white*: pixels ignored in EM)

Fig. 5 **a** CyberGlove, **b** CyberGrasp, and **c** Motionstar wireless tracker

performance. The 350 g CyberGrasp exoskeleton is capable of applying a maximum of 12 N per finger force-feedback at interactive rates and with precise control. The direction of the force feedback is approximately perpendicular to the fingertips.

Additionally to the haptic devices a position tracker device for providing information on the accurate position of the hand is used. Based on the requirements of the proposed application, the MotionStar Wireless Tracker of Ascension Technologies Inc., Burlington, VT, USA has been selected as the appropriate device, mainly due to its wireless nature (Fig. 5c). Combining CyberGrasp with the motion tracker can create a workspace of 6 m diameter hemisphere where the user can move and interact with the virtual model, in contrary with the usual systems that limit the user workspace to be less than half a meter (just in the front of a personal computer).

4.4 Air–mouse interface

The third interface consists of a wireless air–mouse [17] that has the exact functionalities of a typical 2D mouse and can additionally by operated in the air since it utilizes a gyroscope sensor to identify changes in its orientation. Notice that, despite the fact that it can be operated in the 3D space, it is not a 3D mouse.

5 Experimental results

The developed sketch-based 3D search platform was evaluated by 17 users, five 3D designers and 12 researchers and engineers, in many scenarios where the user had to sketch the query object in order to search for similar content. The sketch-based 3D search system was considered to be very innovative and useful especially for the cases where no query model is available that appear very often in practical applications like computer aided design.

Moreover, the users mentioned that the ‘‘Pseudo 3D sketching’’ algorithm was in the beginning easier to use when compared to the ‘‘interactive implicit 3D model sketching.’’ However, after they were introduced to superquadrics and their properties, they managed to use the superquadric-based sketching very efficiently that has the inherent advantage of the very large number of 3D objects that can be sketched.

Finally, the three different interfaces were evaluated with respect to several parameters, which are:

- User immersion.
- Usability.
- 3D manipulation efficiency.
- Mobility.
- Robustness.
- Computational efficiency.

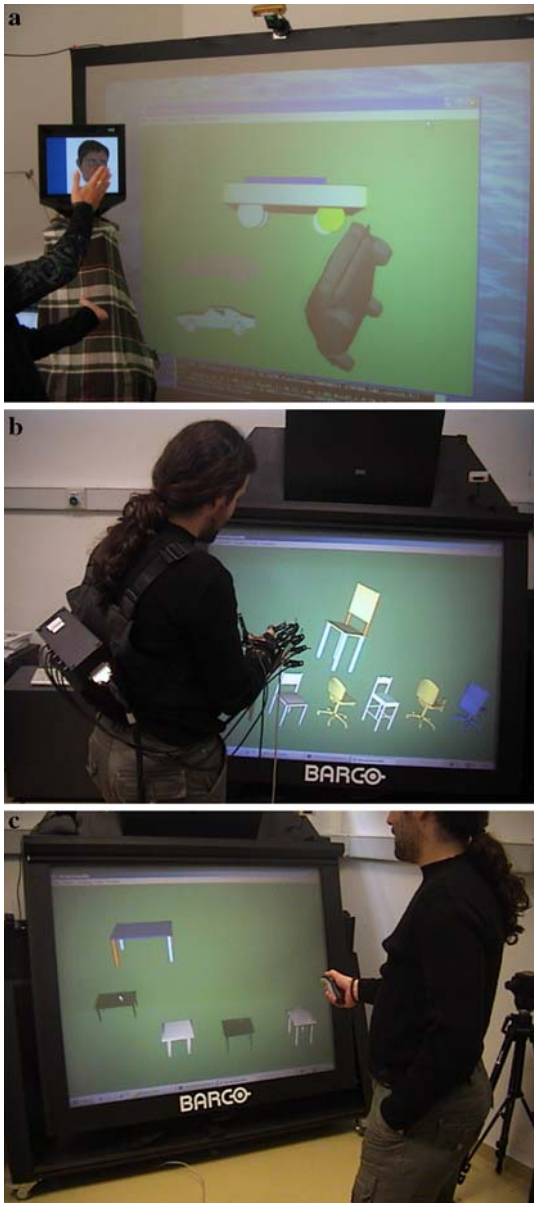

Fig. 6 **a** Unobtrusive interface, **b** Haptic interface, and **c** Air-mouse interface

- Device intrusiveness.
- Cost.

Figure 6 illustrates three snapshots, while using the sketch-based 3D search platform, while Table 2 presents the comparative results of their evaluation.

It is obvious from the above analysis that the most immersive, efficient in 3D manipulation and promising in terms of further development and extension is the unobtrusive interface. The haptic–VR interface has the advantage of providing force feedback, but on the other side the user has to carry relatively heavy equipment and to calibrate the hardware before starting the interaction. Finally, the air–mouse interface is a cheap, portable solution that lacks obviously in user immersion and 3D manipulation efficiency.

Table 2 Comparison of the interfaces

	Unobtrusive	Haptic–VR	Air–mouse
User Immersion	Very high	High	Very low
Usability	Very high	Very high	Moderate
3D manipulation efficiency	Very high	Very high	Very low
Mobility	Very low	Very low	Very high
Robustness	High	Very high	Very high
Computational efficiency	Moderate	High	Very high
Cost	High	Very high	Very Low
Device intrusiveness	Very low	High	Low

6 Conclusions

In the present paper, a sketch-based 3D search system was presented. The user is capable of creating the query object using speech and gesture instead of using an existing model to search for similar 3D content. Three different interfaces for human computer interaction were tested and comparative results were extracted that indicate that each interface has its advantages and disadvantages. Which one to use? It depends absolutely on the context of the application to be developed.

Acknowledgments This work has been conducted in conjunction with the ''SIMILAR'' European Network of Excellence on Multimodal Interfaces of the IST Program of the European Union (http://www.similar.cc) and the ''INTUITION'' European Network of Excellence on Virtual Reality and Virtual Environment Applications for Future Workspaces.

References

1. Berchtold S, Kriegel HP (1997) S3: similarity search in CAD database systems. In: Peckham J (ed) Proceedings of the SIGMOD. ACM, New York, pp 564–567
2. Paquet E, Rioux M (2000) Nefertiti: a tool for 3-D shape databases management. SAE Trans: J Aerosp 108:387–393
3. Loffler J (2000) Content-based retrieval of 3D models in distributed web databases by visual shape information. In: Proceedings of the international conference on information visualisation (IV2000), London, pp 82–87
4. Johnson AE, Hebert M (1999) Using spin-images for efficient multiple model recognition in cluttered 3-D scenes. IEEE Trans Pattern Anal Mach Intell 21(5):433–449
5. Blais G, Levine M (1993) Registering multiview range data to create 3D computer objects. IEEE Trans Pattern Anal Mach Intell 17(8):820–824
6. Chua CS, Jarvis R (1996) 3D Free-form surface registration and object recognition. Int J Comput Vis 17(1):77–99

7. Daras P, Zarpalas D, Tzovaras D, Strintzis MG (2004) 3D model search and retrieval based on the spherical trace transform. In: Proceedings of the IEEE intl workshop on multimedia signal processing, Sienna, Italy
8. Vranic DV, Saupe D (2002) Description of 3D-shape using a complex function on the sphere. In: Proceedings of the IEEE international conference on multimedia and expo, Lausanne, pp 177–180
9. Schwartz R, Chow YL (1990) The N-best algorithm: an efficient and exact procedure for finding the N most likely sentence hypothesis. In: Proceedings of the ICASSP 1990, Albuquerque, pp 81–84
10. Moustakas K, Tzovaras D, Carbini S, Bernier O, Viallet JE, Raidt S, Mancas M, Dimiccoli M, Yagci E, Balci S, Leon EI (2005) MASTER-PIECE: a multimodal (Ges-ture+Speech) interface for 3D model search and retrieval integrated in a virtual assembly application. In: Proceedings of the eNTERFACE 2005, Mons, pp 62–75
11. Marquardt DW (1963) An algorithm for the least-squares estimation of nonlinear parameters. SIAM J Appl Math 11(2):431–441
12. Oliveira M, Colaço V, Jorge J, Fonseca M (2001) Modeling solids and surfaces with sketches: an empirical evaluation. In: Proceedings of the winter school of computer graphics, (WSCG 2001), Plzen, pp 28–31
13. Chevalier L, Jaillet F, Baskurt A (2003) Segmentation and superquadric modeling of 3d objects. J WSCG 11(3):232–240
14. Solina F, Bajcsy R (1990) Recovery of parametric models from range images: the case for superquadrics with global deformations. IEEE Trans Pattern Anal Mach Intell 12(2):131–147
15. Carbini S, Viallet JE, Delphin-Poulat L (2005) Context dependent interpretation of multimodal speech-pointing gesture interface. In: Proceedings of the international conference on multimodal interfaces, Trento, Italy
16. Immersion Technologies Inc. (2000) Virtual hand suite: user and programmer guides. http://www.immersion.com/3d/products/virtualhand_sdk.php
17. Gyration Inc., http://www.gyration.com/go24airmouse.htm