Andrew J. Woods, John O. Merrit, Mark T. Bolas, Ian E. McDowall, SPIE vol 5664, p. 530-539, EI'2005, January 16-20, 2005
in San Jose, (CA - USA)

Reusable methodology based on filters in order to define relevant tangible parts for a TUI

Fabrice Depaulis, Nadine Couture, Jérémy Legardeur, Ludovic Garreau
LIPSI / ESTIA – Technopole Izarbel – 64210 Bidart (France)
{f.depaulis, n.couture, j.legardeur, l.garreau}@estia.fr

ABSTRACT

Modern CAD systems offer many powerful functions to handle parts and assemble them. However, these functions often mask problems that only occur on the final production stage (for example, positioning difficulties for two parts before fixing). The ESKUA project aims to solve this issue by providing a tangible way to test an assembling task, as soon as possible in the design process. In this Tangible User Interface (TUI) based system, each CAD part is associated to a real world object, called interactor. Each action performed with these interactors is captured by a camera, and then visualized in the CAD software.

From a usability point of view, it is very important to provide an appropriate interactor family. This paper deals with a design methodology for such a set.

First, we show how an object can be characterized in the assembling context, regarding a theoretical definition of assembling task. Then, we detail how our methodology gathers together parts that share the same value for a given assembling criterion, and how it builds interactors from this analysis, as abstractions of each subset properties. Finally, we validate the proposed approach with an experimental use to find out an interactors set for mechanical parts assembling.

KEYWORDS

Tangible User Interface, Assembling Design.

1. INTRODUCTION

For many years, CAD systems have been more and more used to design parts and to assemble them. These systems have become more and more powerful. They now provide very high level functions that allow a user to position many parts with only few mouse clicks. For example, it is possible to perform a multiple selection on two, three or more objects with one click. Then, bringing their axis in alignment only requires to select the matching item in a popup menu. Unfortunately, even if these powerful functionalities are very useful from a computer user point of view, they mask real problems that only occur in the final production stage. In such a situation, the operator may not have hands enough to handle all the parts and to perform the alignment.

As a conclusion, there is obviously a wide gap between the way this action is performed in the CAD system, and the way it is in the real word. This gap cooperates in hiding positioning difficulties for two parts before fixing, but also parts insertion problems such as collisions.

We think that the handling of real objects makes it possible to “anticipate” some physical aspects of the product assembly phase and leads the designer to raise questions in a “natural” way by carrying out the gestures related to the assembly. Moreover, it is now admitted [8] that the use of real objects for displacements and the control of the virtual objects is more powerful than the traditional systems such as 3D mouse. This have led us to consider a new TUI dedicated to this specific application field.

For two years, we have been working on a new way to plan assembling tasks in the earliest stages of the product design.

The project, called ESKUA [2][3] (stands for “Expérimentation d'un Système Kinésique Utilisable pour l'Assemblage”[1]), consists in developing a system that completes CAD software with the emerging paradigm of Tangible User Interface (or TUI).

Since 1979 and Aish tangible interface for handling a building structure [1], many TUI based systems proposed to use real world objects as interaction devices. The term “TUI” comes from an analogy to GUI (Graphical User Interface) and has been described as a physical realization of a GUI in [7]. To refer to these objects, we use the term “Interactor”. It comes from “Interaction” and “Actor”

The ESKUA system is composed of a platform, a video camera (Figure 1) and a set of interactors (Figure 2). The user first associates each CAD object with a given interactor; then, he can perform the assembling task in the real world with these interactors. Thanks to the previous association and video tracking algorithms, each action is displayed in the CAD system.

Contrariwise to sophisticated projects, based on Virtual Reality techniques [5], our system is cheap (the price of the first prototype is less than 1500 €), doesn't require any expensive high technological device (as headmounted displays), and is easily integrable in the designer workspace.

Figure 1. First ESKUA prototype.

Figure 2. First interactor set prototype

Beside the video acquisition and the move tracking, one of the rising major problem of this approach is the difficulty of designing a “valid” interactor set - a set that fits the domain. Indeed, so as to be able to represent real problems, a chosen generic form has to be a faithfully representation of a real part, from the assembling point of view. Moreover, as our aim is not to prototype but to process simulations, we don't want each interactor to correspond to a catalogue real part of the domain.

The challenge of this paper is to exhibit a method that finds out a valid interactor set, relatively to an assembling process and a given application area. Starting from the previous observations, the article first deals with a theoretical study of assembling task, based on Rejneri definitions [6]. It confirms that, in the context of an assembling process, an object can be fully represented by its functional surfaces: planes, cylinders, cones and spheres. This preliminary work exhibits criteria that characterize an object in the context of assembling. Then, we explain our approach, which consists in categorizing a relevant part set of the addressed area, compared with these theoretical criteria, and to identify the matching interactor set from successive filtering. Finally, in order to validate the proposed approach, we present a test of our methodology, showing how it has been used to find out an interactor set for mechanical part assembling.

1French acronym that means “experiment of a usable tangible system for assembling”.

2. ASSEMBLING TASK ANALYSE

In a previous work [3], we focused on DFA (Design For Assembly) methods (presented for example in [4]). So as to evaluate the assembling difficulties of a product during the design phase, they recommend to provide a number of required data, following several steps. From this methods, we designed a first interactor set and tested it with different type of user: designers, assembling experts, CAD users and ergonomic experts. This first experiment led us to conclude that a user represents a part with interactors using two main criteria: the general object form and its functional surfaces.

In this section, we define an assembling task from the degrees of freedom of an object that takes part in the process (most of the following definitions are from [6][9]).

2.1 Degrees of Freedom of a Rigid Body in Space

A rigid body is a body with points at a constant distance from each other during a movement. The position of free moving solid is set by six parameters: a point position (three coordinates) and three Euler angles. In the most general case of movement, the solid owns six freedom degrees. An unrestrained rigid body in space has six degrees of freedom: three translating motions along the *x*, *y* and *z* axes and three rotary motions around the *x*, *y* and *z* axes respectively (Figure 3).

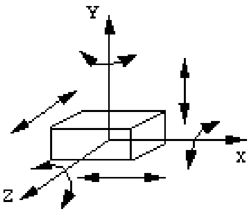


Figure 3. Degrees of freedom of a rigid body in space.

2.2 Kinematic Constraints

Two or more rigid bodies in space are collectively called a *rigid body system*. We can hinder the motion of these independent rigid bodies with *kinematic constraints*. *Kinematic constraints* are constraints between rigid bodies that result in the decrease of the degrees of freedom of rigid body system.

The term kinematic pairs actually refers to *kinematic constraints* between rigid bodies. The kinematic pairs are divided into lower pairs and higher pairs, depending on how the two bodies are in contact. Each pair corresponds to a number of degrees of freedom.

For example, a planar pair refers to one rotary and two translating moves (i.e. three degrees of freedom), a revolute pair refers to only one translating move, etc. In France, this pairs are known under the name of "links" and are normalized (NF E 04-015). The Figure 4 presents these links, in regards to the degrees of freedom (number of translations and rotaries) they offer.

2.3 Assembling by degrees of freedom

Calculating the degrees of freedom of a rigid body system is straight forward. Any unconstrained rigid body has six degrees of freedom in space and three degrees of freedom in a plane. Adding kinematic constraints between rigid bodies will correspondingly decrease the degrees of freedom of the rigid body system.

According to Nicolas Rejnerii [6], any object can be considered as built from the elementary components that are parts of its structure, by drawing up the links between each other. In an assembling task context, a relation means that there is a contact between the components. This contact is completed by a cohesion energy brought by specific elements (screw, rivet, ...) or by an other constraint (solder, glue, ...).

Consequently, an assembling is entirely defined by the basic components that constitute the final part, and by the joints between each other. As we are able to represent all the possible joints between two parts, we are able to represent all the assembling tasks too - note that this conclusion considers an assembling as relations between components and doesn't take into account geometric or trajectory problems.

The next part of this paper starts from this definition so as to find out a set of geometrical forms that would allow to represent any assembling process. This set is "theoretical" as it is based on a theoretical assembling definition.

3. A THEORETICAL INTERACTOR SET FOR ASSEMBLING

According to the previous part of this paper, an assembling can be entirely defined by its basic components and the joints that link them to each other. Thus, finding out the forms that will allow to simulate all the assembling processes consists in finding out the forms that will allow to represent all these joints. This section shows which surfaces, and which 3D forms are required to represent all kind of kinematic pairs.

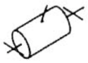
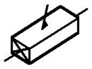
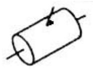
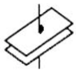
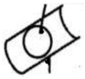
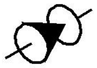
Pair	**#translating / #rotary**	**Representation**
Rigid joint	0 t / 0 r	
Revolute	0t / 1r	
Prismatic	1t / 0r	
Cylindrical	1t / 1r	
Planar	2t / 1r	
Spherical	0t / 3r	
Annular linear	1t / 3r	
Rectilinear linear	2t / 2r	
One-off	2t / 3t	

Figure 4. NF E 04-015 kinematic pairs (pairs mentioned in an italic font are direct translations of French names and may not correspond to the English notations)

3.1 Part surfaces and pairs

For Rejneri, the link between two components is entirely characterised by the type of contact. To understand this contact and the resulting degrees of freedom, the involved functional surfaces have to be known. A functional surface is a part surface that owns a geometric interaction with surfaces from other parts. There are four elementary functional surfaces: plan, cylinder, sphere and cone. Getting two surfaces in contact defines a kinematic pair. For example, getting two planar surfaces in contact creates a planar pair, owning three degrees of freedom (one rotary and two translations). These resulting pairs are resumed in the Figure 5 (note that pairs mentioned in an italic font are literal translation of French terms and may not refer to the classical English language notation).

This table shows that it is possible to represent all the NF E 04-015 kinematic pairs with four basic functional surfaces. Consequently, if we are able to create geometric 3D forms that represents all these surfaces, we will be able to create all the basic kinematic pairs, and then to build any assembly.

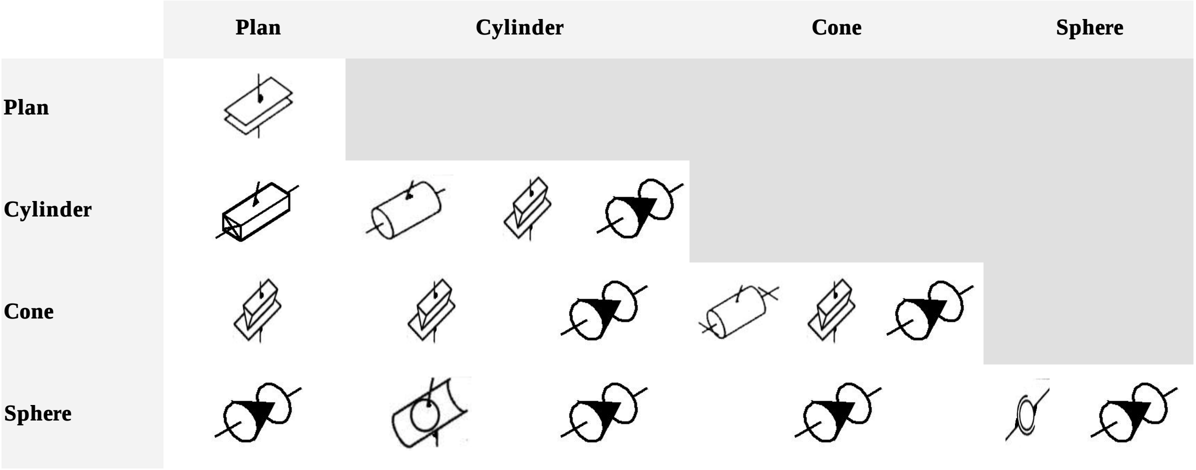


Figure 5. Kinematic pairs resulting from getting in contact two functional surfaces.

3.2 Functional surface based interactor set

As a conclusion, an interactor set that allows to get in contact any of the four basic functional surfaces, will allow to create all the basic kinematic pairs and to build any assembly. Thus, such a set would be constituted with parallelepipeds, cylinders, cones and spheres (Figure 6).

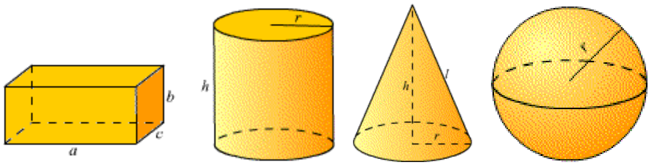


Figure 6. Basic geometric forms corresponding to the theoretical interactor set.

It is a minimal set based on elementary geometric forms. Nevertheless, each interactor that represents such a form contains one ore more functional surfaces. For example, a parallelepiped owns six functional surfaces (plans), a cylinder owns two functional surfaces (cylinders and plans), etc.

This leads to a first proposal:

> *Let A = {A1; A2; ..., An} the whole set of all the assemblies, where Ai is an assembly.*
>
> *Let I = {I1, I2, I3, I4} an interactor set where I1, I2, I3 and I4 respectively represents a cube, a cylinder, a cone and a sphere (independently of their sizes).*
>
> *Combining several elements of I makes it possible to create a representation of any component, with respect to its functional surfaces. As an element of A can be represented the parts involved in A, and the joints between them, and since all the joints can be represented with the functional surfaces of I elements; any Ai from A can be represented with a combination of I elements, independently from fixation elements.*

This first proposal seems to prove that the problems is very easy to solve: if we summarize an assembly to the joints between the parts involved, it is possible to represent all the assemblies with a very restricted interactor set. Nevertheless, as we plan to design a platform for a practical use, we have to take into account usability parameters. The next part of this paper shows why this solution doesn't fit our needs.

3.3 Theoretical interactor set usability

Mechanical part assembling involves basic components that are classified (21th section of the ICS : mechanical systems and components of general use) and ISO normalised. From this normalization, we can get part families used in most assembling tasks.

So as to test the usability of our theoretical interactor set, we have used our interactor set to model a mechanical part. We have chosen to study a clamp (Figure 7), which belongs to the normalized mechanical part families and which is used in numerous mechanical assembling tasks.

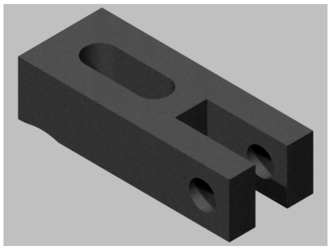


Figure 7. A clamp

The clamp can be characterized by its functional surfaces: ten plans and four cylinders. So as to model this part with our interactor set, in respect with its functional surfaces, we have to achieve a complex assembling from eight parallelepipeds and four cylinders (twelve interactors). This requires a great effort from the user only to create a very simple assembling element (maybe will he/she need two or three clamps for each assembling task simulation). This is a preliminary operation without any interest for the process he/she wants to simulate, that could be much harder to achieve than the process itself.

In this case, using two more specific interactor would significantly increase the usability of the system. For example, the preliminary work would be simplified if the interactor sets contained two more sophisticated components (Figure 8).

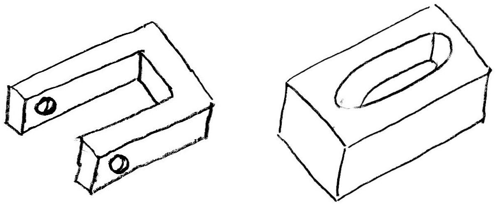


Figure 8. Association of two more specific interactors to make a clamp.

3.4 Conclusion on the theoretical interactor set

We started from a specific assembly definition and find out a "theoretical" interator set that owns only four elements, and that allows to simulate any assembling task, in respect with the real objects functional surfaces. Unfortunately, this approach doesn't fit the need of a particular and well defined domain. The granularity level and the basic components depends of the addressed area: a system that aims to help a user in assembling a motor can't require the user to make his/her own crank arms.

Consequently, it is not possible to design an interactor set without studying the concerned application field, since each domain has a different granularity level and uses different kind of basic components. On the other hand, we can't use an interactor set that would be an exact copy of the basic elements of the addressed domain. Indeed, we don't want to build a system to perform prototyping, but to make the designer understand assembling problems. For this reason, we want to use an abstraction of the basic elements, defined from relevant assembly properties, that would be enough for solving the problem. Moreover, an exhaustive set would make the system unusable - there are more than 600 mechanical parts defined in the 21th section of the ICS.

The next part of this paper deals with another approach. It consists in finding out part properties that are relevant for an assembling task. Then, from a representative and relevant part set (the most used in assembling for a given domain), a methodology is given, that consists in gathering together the basic elements which share the same values for these characteristics.

4. AN EXAMPLE BASED INTERACTOR SET FOR ASSEMBLY

4.1 A mixed approach

Our methodology is based on the previous theoretical analyse, and uses the functional surfaces of the basic components of a domain. But, contrariwise to the previous approach, it also takes domain properties into account to design the interactors.

The preamble to this methodology consists in identifying criteria that make a component conspicuous in the particular context of assembly. This part of the work starts from the analyse led on the functional surfaces. Then the basic component set of the addressed domain has to be found out. These elements must be the "most" used in the common assembly processes, and require the point of view of an area expert. Finally, this set is analysed: each component gives values for the identified criteria, and is gathered with objects sharing the same values. Each created family correspond to an interactor.

4.2 Classification criteria

So as to define the component equivalence for an assembly task, it is important to choose the "minimal" criterion set. Indeed, the more numerous the criteria are, the more numerous the families will be: now, we don't want the interactor set to exactly match with the basic component set.

For this reason, we present an incremental approach. This lead to three proposals, each adding a new criterion to the previous one. Note that even if we try to adpot

The first proposal is based on the conclusion of the first part of this paper. It argues that an object involved in an assembling process can be defined by the number and the type of its functional surfaces.

> *Proposal n°1*
>
> *Let St the set containing all the functional surfaces that can be a part of an object O. St = {p,s,co,cy} where p is the planar surface, s the spherical surface, co the conic surface, cy the cylindrical surface.*
>
> *An object O is defined by a <Po, So, CYo, COo> where Po is the set of the planar surfaces of O, So the set of the spherical surfaces of O, CYo the set of the cylindrical surface, COo the conic surface.*
>
> *Two objects Oi and Oj are equivalent in an assembling process if and only if Cardinal(Poi)=Cardinal(Poj) AND Cardinal(Soi)=Cardinal(Soj) AND Cardinal(CYoi)=Cardinal(CYoj) AND Cardinal(COoi)=Cardinal(COoj)*

Obviously, this proposal doesn't fit our needs. We can easily find out two objects owning the same number of each type of functional surfaces and which are not equals for an assembly. For example, the two objects of the Figure 9 owns exactly the same sets P, CY, CO and S, and would obviously not be equivalent in an assembly process.

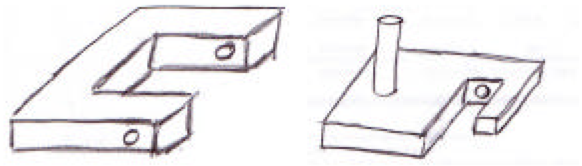


Figure 9. Equivalent objects for proposal n°1.

So as to solve this problem, the second proposal takes the surface normal vector into account.

> *Proposal n°2*

Let St the set containing all the functional surfaces that can be a part of an object O. St = {p,s,co,cy} where p is the planar surface, s the spherical surface, co the conic surface, cy the cylindrical surface.

An object O is defined by the set Co of the couples (Sti, ni), where Sti is a functional surface that belongs to St, and ni is a Sti normal, oriented toward the matter.

Two objects Oi and Oj are equivalent for an assembly task if and only if Coi=Coj

These new criteria neither fill our needs, as they don't define the orientation of the revolution surfaces. Thus, the three objects of the Figure 10 are equals for the proposal n°2 but obviously not for an assembly task.

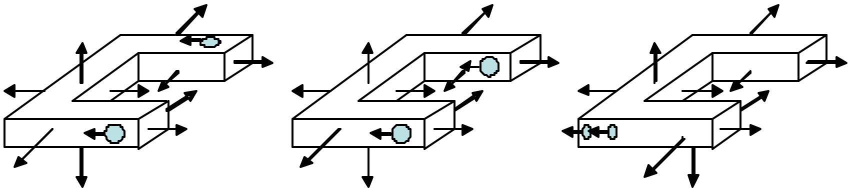


Figure 10. Equivalent objects for proposal n°2.

So, we have to consider both the number and the type of the functional surfaces, the direction of their normal and the position and direction of the axe of the revolution surfaces. This leads to the third proposal.

Proposal n°3

Let St the set containing all the functional surfaces that can be a part of an object O. St = {p,s,co,cy} where p is the planar surface, s the spherical surface, co the conic surface, cy the cylindrical surface.

An object O is defined by the set Co of the 3-uplets (Sti, ni, ai), where Sti is a functional surface that belongs to St, ni is a normal vector to Sti, oriented toward the matter, and ai is the axe of the revolution surfaces.

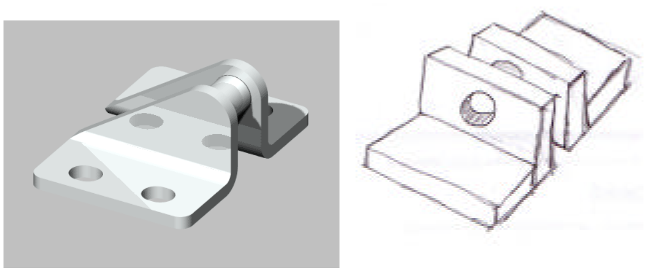
Two objects Oi and Oj are equals in an assembly task if and only if Coi=Coj.

This proposal seems to be a "relatively good" compromise, but it is not possible to demonstrate it formally.

4.3 Methodology

The previous part shows two objects can be considered as equals if they own the same functional surfaces (number and type), if the surface normals are the same and if the revolution surface axis have the same direction. From these criteria, a classification of a basic component set can be done: gathering the objects that share the same value (for each criterion) allows to build component families, that can be used to create the interactors. The so built interactors fit exactly the assembly in this particular domain.

It is important to notice that, during the analyse phase, only functional surfaces useful for assembly have to be taken into account: surfaces that will never be in contact with other surfaces have to be ignored. For example, the representation of the mechanical part of the Figure 11 doesn't take into account details as chamfers.



(a) (b)

Figure 11. (a) Real mechanical part. (b) Representation based on the functional surfaces.

To summarize, the methodology steps are:

1. Get a relevant basic component set of the domain. This set must contains the most used elements, in the assembly of the addressed area.
2. Categorize each element of the set, in regards to the exhibited criteria: number and type of functional surfaces, direction of the surfaces normal, direction of the revolution surface axis.
3. Gather together all the component that share the same value for the criteria. This gives component families from which interactors will be extracted: their shape design is led by the criteria values. Thus, each interactor is an abstraction of a whole set of components, in regards to their assembly properties.

At this step, the found out families might be too numerous again. Then, we recommend to filter the interactors with a loop on the set: "while the number of the interactors is greater than 20, remove the interactors that are a combination of n interactors" (n starting value is 2, and is increased of one at each iteration). It seems that a set of twenty elements is a good compromise to keep the system usable, even if this value hasn't be experimentally proved.

4.4 Experimental test

So as to test this methodology, we have applied it to the particular domain of mechanical part assembly. For this experimentation, we have chosen to use the ISO normalized components (21th section of the ICS : mechanical systems and components of general use). They are more than 600 and are relevant of the basic components used in mechanical assembly.

The Figure 12 shows that filtering the first set is required, as it contains 61 elements.

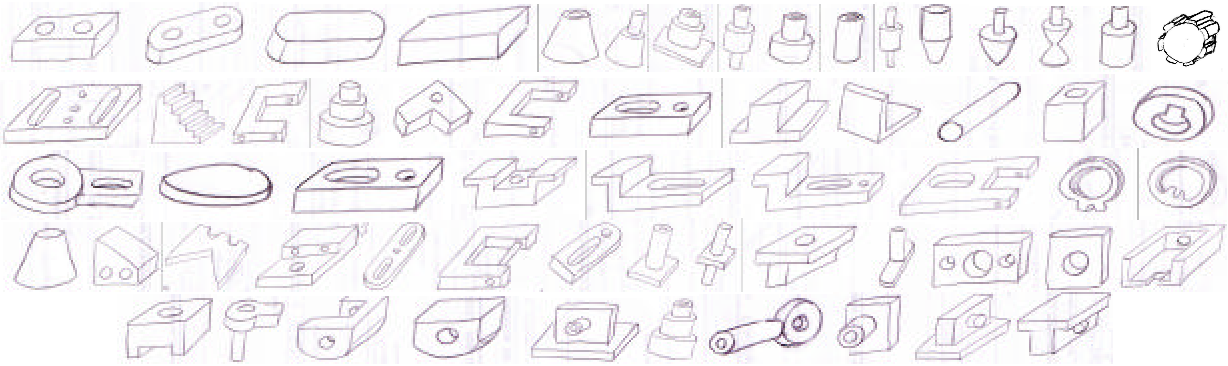


Figure 12. The first interactor set (61 elements)

The Figure 13 shows that filtering several times the initial set allows to decrease the number of elements.

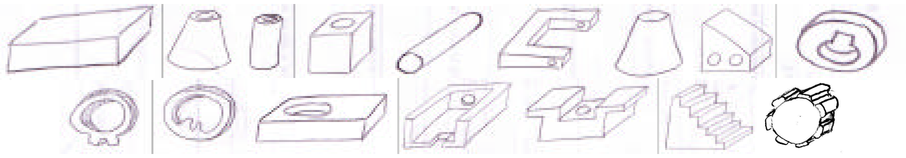


Figure 13. The final interactor set (16 elements)

We plan to build these interactors, to test them in a real context and to compare the result with the use of the first set we have realized one year ago.

5. CONCLUSION AND PERSPECTIVE

Using a Tangible User Interface so as to bring real assembly problems in the designer office is an exciting challenge that deals both with assembly theory, video tracking, Human-Computer Interaction, etc. But using real object to interact with the numerical world brings new questions and especially: what is the most relevant object to achieve the user task ? This question may have different answers, according to the addressed area of the system.

In this paper, we have shown that it was possible to find out the "best" interactor set for performing an assembly in a given domain. These interactors are both representative of the basic components used in the area, and few enough to keep the system usable. Indeed, we saw that a purely theoretical set, based on a simplified definition of assembly, leads to an unusable solution. The main reason is that the interactors have to rely to the addressed domain. So, we expressed a new design approach, that takes the domain into account. The method consists in analysing a relevant component set of the domain, and to class them in regards to generic criteria from theoretical definitions. Finally, we applied our methodology to the domain of mechanics, and find out an interactor set containing 16 elements, from an initial part set containing more than 600 ones.

This work is a part of a wider project that includes several further challenges. For example, we plan to experiment this new interactor set, and to compare it with our previous one, with real users. This evaluation may involve real designers and mechanics experts. Moreover we want to extend our platform so as to represent the cohesion elements that are involved in an assembly process. We plan to use a similar approach to find out the relevant "cohesion interactors".

REFERENCES

1. R. Aish, *3D Input for CAD Systems*, . Computer Aided Design, volume 11, n°2, 1979.
2. L. Garreau, J. Legardeur, N. Couture, *Tangible Interface for mechanical CAD parts assembly*, In Proc. Virtual Concept 2003, p. 222-227, VC'2003, November 5-7, 2003, Biarritz (France).
3. J. Legardeur, L. Garreau, N. Couture, *Experiments to evolve toward a tangible user interface for CAD parts assembly*, in Proc. Electronic Imaging Science and Technology, Stereoscopic Displays and Virtual Relality Systems XI, Editors: Andrew J. Woods, John O. Merrit, Stephen A. Benton, Mark T. Bolas, SPIE vol 5291, p. 438-445, EI'2004, January 18-22, 2004, San Jose (CA - USA).
4. J. Legardeur, *Prise en compte des contraintes d'assemblage dans une logique de conception intégrée : l'approche Design For Assembly*, Master Report in Mechanics and Design, Université Joseph Fourier, Grenoble, 1998.
5. Perf RV, European Project, *Virtual Reality Experimentation Center for the Future Designer Office*, http://www.perfrv.org/.
6. N. Rejneri, *Détermination et simulation des opérations d'assemblage lors de la conception de systèmes mécaniques*, PhD. Thesis, Institut National Polytechnique de Grenoble, 2000.
7. B. Ullmer, and H. Ishii, *Emerging Frameworks for tangible user interfaces*, Human Computer Interaction in the New Millenium, pages 579-601, 2001.
8. S. Zhai, and P. Milgram, *Human Performance Evaluation of Manipulation Schemes in Virtual Environments*. In Proceedings of IEEE Virtual Reality Annual Symposium (VRAIS'93), pages 155-161, 1993.
9. Y. Zhang, *Index of Mechanism Terminologies*, On line Carnegie Mellon University Report, http://www-2.cs.cmu.edu/People/rapidproto/mechanisms/.