Proceedings of DETC’01
2001 ASME Design Engineering Technical Conferences
And Computers and Information in Engineering Conference
Pittsburgh, Pennsylvania, September 9-12, 2001

DETC’01/DTM-21708

A methodology for reusing freeform shape content

Joris S.M. Vergeest
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
j.s.m.vergeest@io.tudelft.nl

Imre Horváth
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
i.horváth@io.tudelft.nl

Sander Spanjaard
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
s.spanjaard@its.tudelft.nl

ABSTRACT

The reuse of precedent designs is a significant profit factor in new product development. In industry there is a tendency to enhance the reuse process by applying digital scanning of 3D parts, sampling imported, nonnative CAD models or by deploying a digital library of design concepts. The data thus obtained should be inserted into the design model. The available techniques typically originate from reverse engineering applications. However, to support shape reuse during conceptual design a dedicated methodology and workflow are needed. Using our methodology, the designer selects existing products, or parts, or portions of them. Then he/she specifies where and how the selected portion should be inserted into the new design. The key issue of the methodology is the explicit distinction between the variables that the designer does or does not wants to control. The underlying technology, including shape matching, shape parameter fitting and shape merging must be mostly invisible to the user, except for those controls that intrinsically affect the resulting shape. One application of the methodology is a freeform feature copy-and-paste facility based on 3D scanning and fitting of existent designs. The technical feasibility of such an approach will be addressed.

INTRODUCTION

In new product development two trends are becoming of dominating importance, 1) enhancement of freeform shape design and its accomplishment using *freeform features*, 2) the integration of virtual and physical modeling in the freeform domain, which implicitly calls for *shape reuse* methods.

Freeform features

Especially for consumer products the outer appearance represents a key value of the design. Despite the considerable proliferation of tools for computer-aided industrial design (CAID), the creation, manipulation and management of freeform objects is still only marginally supported in comparison to the impact exerted by *e.g.* mechanical CAD systems. Nevertheless, the expectations regarding the product’s aesthetic and ergonomic quality are very high, and design proposals are to be delivered under severe time pressure. This led to a number of attempts to apply principles of feature-based design from mechanical design in the domain of freeform shape modeling and styling design. Recently, (Au 2000) proposed a grammatical feature modeling to aggregate non-regular shaped features into a sculptured object. The features and their relations to the object and to other features formed the vocabulary of a feature language. The language was partially object-specific and explicitly contained the hierarchical structure of the design and rules for connecting spatially adjacent features. The final geometry of the individual feature instances was derived by solving the constraints implied by the rules. The system proposed by Au *et al.* enables the designer to quickly generate object variations within a given framework of the object’s structure. The object specificity, however, appears as a limitation of the feature approach to freeform object modeling. On the other hand it has been shown in (Mitchell 2000) that an object-specific feature anatomy enables accurate performance predictions of the product design.

A big contrast between regular-shaped features and freeform features is that the latter can hardly be predefined generically, but should evolve in a specific design context and be customized consequently. This seems to call for a different design workflow than the one known for mechanical feature-based design. Especially the principle of aggregation of feature instances as to form the entire design model seems very restrictive. This may explain the considerable attention that has been payed to the development of detail features modeling techniques, as for example by (Cavendish 1995) and (van Elsas 1998). Also the proposal for the classification of freeform features by (Fontana 1999) emphasizes the detailed features over the structural features. In (Poldermann 1995), a distinction is made between global and local surface features. Alternative freeform feature classifications have been proposed by (Gindy 1989) and (Eversheim 2000), both aiming at the incorporation of manufacturing aspects into the designed shape.

Another distinction between regular-shapes and freeform features is their spatial boundary, which is very clear for steps, holes, slots and the like but sometimes quite ambiguous for styling lines, smooth cavities etc. Perhaps, surface features should be treated in a dual way: both as a set of shape handles (Hsu 1992) and as constituents of an object.

Shape reuse

Shape reuse is the application of precedent modeling effort in the current design model. It is a specific kind of design reuse, which is generally appreciated as a profit factor in new product design (Duffy 1999). The most commonly known form of shape reuse is the usage of digital part libraries for CAD. Especially when the "standard parts" are parameterized models, a significant reduction of design effort is achieved (De Martino 1994). However, the success is limited to frequently recurring types of shapes, which are typically regular shaped. It is therefore challenging to enable the reuse of shapes beyond the domain of regular shapes. To achieve this, at least three key issues need to be resolved: 1) the creation of new freeform shape types (*i.e.* features) at runtime, as opposed to the formation of a hardwired library of standard parts, 2) finding a parameterization scheme for freeform features, 3) the development and verification of an efficient workflow incorporating design by reuse. It depends on this latter issue how to approach the problem of user-defined freeform features and how to parameterize them. The requirements from the end-user (*i.e.* from the designer or stylist using the envisaged system) determine which techniques known from mechanical feature design can be adopted and which cannot.

Those requirements are addressed in the following section. From these requirements we derive a methodology and workflow for design by reuse in the freeform shape domain. Then, we review the key elements of the methodology against their computational feasibility and we present some examples of solved subproblems.

.

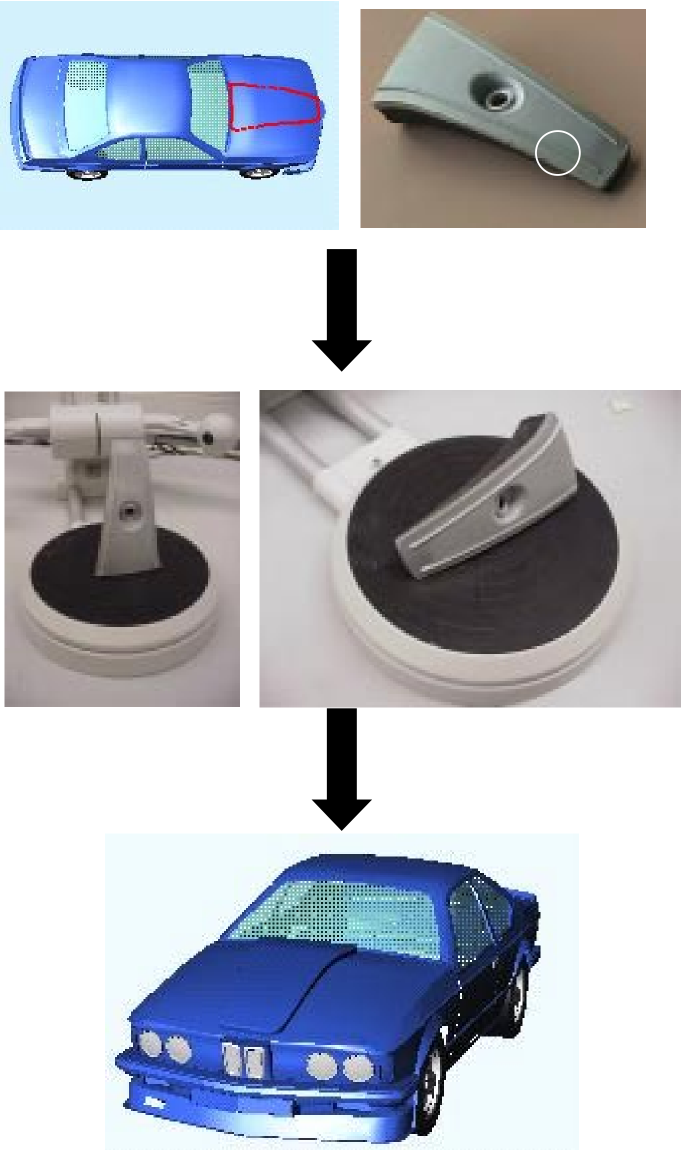

Figure 1. Simplified depiction of the workflow of shape reuse. Top: the designer intends to include a ridge to the car motor hood, where the ridge should be similar to the one observed in the object (see circle). The ridge is scanned using a desktop laser scanner and the ridge is then inserted into the CAD model of the car.

REQUIREMENTS FOR SHAPE REUSE IN CONCEPTUAL DESIGN

The scope of the methodology is the explicit insertion of an encountered shape feature into the current design, a simple example is depicted in figure 1. Possible sources of the features include physical objects (which then need to be scanned) or portions of 3D digital models, where a model can be fully nonnative (*e.g.* found on the Internet) or it can be the current design itself, to mention two extremes

The principle functions that should be provided according to the methodology are:

1. A selection method that enables the user to designate existing shapes or portions of them. The existing shapes can be either boundaries of physical objects, or shape models. Shape models include physical 2D sketches, but most commonly they will be available in the form of computer representations. The selected portions are intended for reuse.
2. A conversion method that creates a computer-based representation of the selected shape. In case the selected shape is encountered on a physical object, a process of 3D digitizing and surface reconstruction will be involved. In case the selected shape is already available in digital form, then only the surface reconstruction is needed. Only incidentally, the original surface representation will be adequate for the purpose of reusing the shape in the destination model. Therefore, generally, a representation conversion will be needed.
3. A selection method that enables the user to describe the shape-to-be-reused as an entity that can be manipulated. Only incidentally the selected shape has precisely the geometry as required in the destination model, even when (due to requirement 2) the internal representation is of the right type. Generally, some modifications of the shape are needed before it is inserted into the destination shape. In the case of freeform features, these modifications are typically applied by using parameters. The selection comprises the specification of a type of shape, or the feature type, that should be applied to the selected shape. These types can be either predefined or be user-defined.
4. The internal representation of the selected shape should be upgraded to the representation implied by the chosen feature type. This involves the matching of the shape (either its available representation, or the originally measured points) to synthetic shapes of the given feature type. If a match is found then the selected shape is available as an instance of the feature type, and hence ready for the user to be controlled by its defining parameters.
5. The user must be able to revise his/her choice of feature type. If during the reuse process it appears that a particular type of manipulation is not supported by the system, *i.e.* was not anticipated when the type of feature, and hence its parameterization, was selected, then either the new parameter should be included in the set of parameters, or the feature type should be revised. This involves the technique mentioned in requirement 4.
6. The user must be able to evaluate the result of insertion of

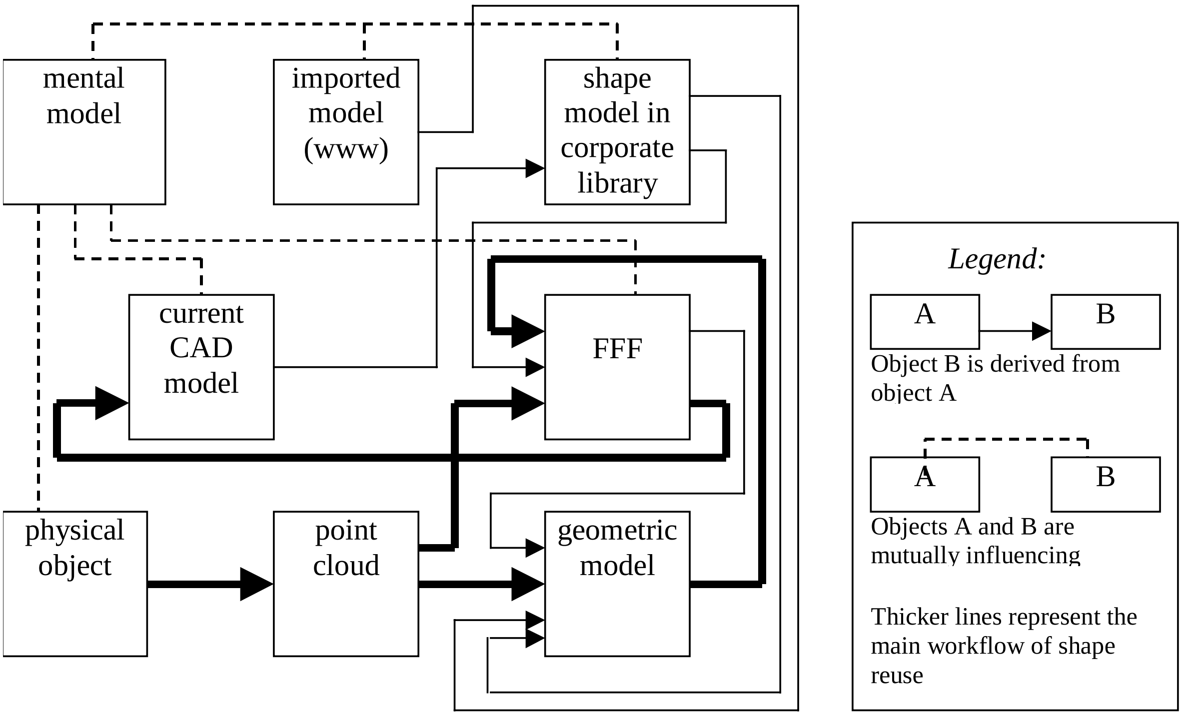

Figure 2. The eight principle objects relevant in the shape reuse methodology. The lines and arrows represent dependencies among the objects.

the shape feature into the destination shape instantly during his/her controlling the feature parameters. This is to adequately merge the feature with the new design. Placement and scaling control are extremely important, as will be particular shape manipulations, permitted by the feature definition.

7. There should be automatic fine tuning for merge of the feature with the destination shape so that the transition between the two entities conforms to smoothness criteria. These criteria are usually defaulted to geometric continuity conditions, but they may be user-defined as well.
8. A facility should be provided to store all user-defined feature types for later reference. In this way the system expands its library of available shape parameterizations.

Requirement 5 addresses a key challenge of contemporary CAD research. If a designer changes his/her mind about a once decided type of shape parameterization, then it is highly untrivial to actually re-represent the shape. Yet, the flexibility to switch among parameterizations would enormously increase the computer-supported design performance.

DESCRIPTION OF THE METHODOLOGY

To describe the shape reuse methodology complying to the above requirements we need to account for the following objects depicted in figure 1.

Current CAD model. This is the presumed design model at a particular stage of development. It is typically a computer-based model in one of the representation known for CAD system, *e.g.* a boundary representation and/or a model built of freeform surfaces. The precise type of representation is of no relevance to the methodology (of course it will be to its technical implementation, as discussed in the next section). The current CAD model is understood as the model of most relevance to the designer (or design team). All other objects can be regarded as auxiliary to the current CAD model (although in a particular implementation some of the models may be available on one system, or even be integrated).

Physical object. This can be any observable thing. Typically it is a manmade product, for example a desk light, a telephone set, a car, or a clay model. If the object is tangible then it can be digitized by a 3D scanning system. If the physical object contains a portion having a shape that could be used in the next step of development of the current CAD model, then the physical object is relevant in the shape reuse methodology.

Point cloud. The result of object scanning is a point cloud, a set of measured locations on the surface of the physical object. Depending on the applied scanning technology, the points may be implicitly ordered into grid structures or the like, but they can be totally unordered as well.

Geometric model. The geometric model can be constructed from a point cloud, as is common practice in reverse engineering applications (Várady 1997). Typically, the geometric model is represented by a set of surfaces (with or without connectivity information) or as a B-rep. It can be considered subservient to the current CAD model (defined above). The geometric model may be a temporary representation of a portion of a shape, which can later be inserted into the current CAD model. In some implementations the geometric model and the current CAD model are identical.

Freeform Feature (FFF). This is a shape specified by a (multidimensional) parameter value; an instance generated from a feature type and the parameter value.

Imported model. Any shape model retrieved from the internet or sent by a partner in the design process. Possibly, the representation of the imported model is different from any of the representation forms known by the receiving CAD system.

Shape model in corporate library. Corporate library represents a repository of shape models which is local to the user or the company. A common implementation form is a company-specific parts library, but it can also take the form of a database private to a CAD system's user. The shape models in the library can be any of the types *point cloud, geometric model, imported model, current CAD model* or *FFF*.

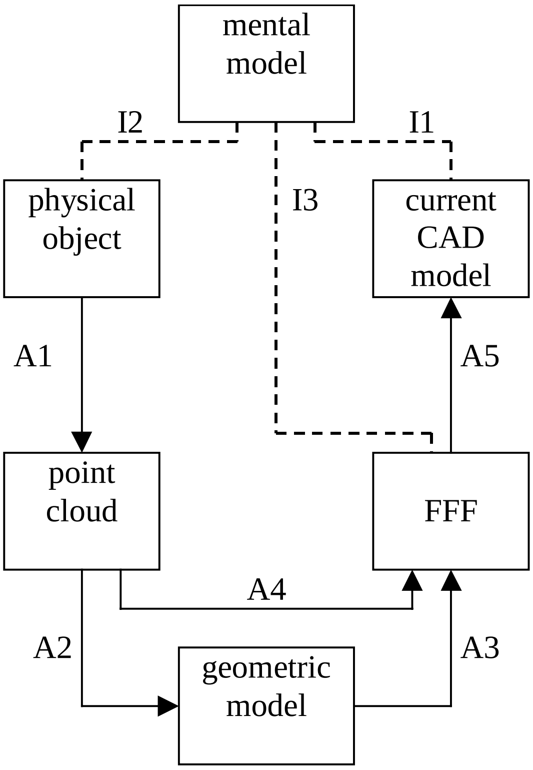

Figure 3. Basic cycle of the shape reuse process. For the legend we refer to Figure 2.

Mental model. This is an identifier for the interface between the human designer and the otherwise computer-based objects. The mental model is then a presupposed image or understanding that a subject has about the designed shape. We will assume particular forms of interplay between the mental model and the exteriorly presented shape (Goldschmidt 1991).

Using the eight entities just defined, the workflow of shape reuse can be described on various levels of detail. On a coarse level only the thick arrows in figure 2 are relevant, and the design process shown in figure 3 is implied. This picture is an oversimplification; it emphasizes the actual information flow inherent to the shape reuse process. As mentioned, physical object, for example, can be replaced by imported model, without invalidating the picture.

The user interacts with three objects, the current CAD model, the physical object and the FFF, indicated in figure 3 by the influences I1, I2 and I3, respectively. The object types have been defined above. I1 is an identifier of what is commonly understood by computer-aided design in any of its forms. In this paper we assume that a user interactively manipulates the current CAD model, part of which represents the geometric shape of the design. On different levels of abstraction the user may wish to reuse precedent designs or shapes. It can be on an informal level, where the designer applies methods or actions that he/she recalls from past experience or learnt strategies. On a more concrete level, the designer may want to, or is requested to, apply particular archetypes or patterns into the new design, for example to comply with product's brand styling guidelines (Smyth 2000). On this level, influence I2 enters the methodology when the user refers to a physical model to obtain a starting point for form evolution, although the starting form can also be created explicitly (using I1) or by any other input means. In Smyth's approach shape concept generation starts from a skeleton archetype which then evolves and mutates through some parameter space; the user selects the most appropriate instance. When shape reuse has a spatially local context, then the embedding of the new shape feature into the current design is crucial (this is activity A5 addressed later). This embedding may be partially automated but in general will involve some attention of the designer; this influence is denoted I3. In an even more concrete manner there are at least two other possible scenarios, 1) the designer encounters a physical (or digital) object containing a shape feature that he/she believes is useful in the current design, or 2) the designer searches among physical objects with the aim to borrow some or all of its shape for inclusion into the current design. These scenarios involve human search, recognition and judgement, in figure 3 modeled by the influences I1, I2 and I3.

A fundamental factor in shape reuse is the adaptation of the reused feature in the current design. We already mentioned the geometric embedding of the feature, but also its shape might need significant modification, which should be supported by a dedicated parameterization of the feature. Which parameters need to be regulated depends on the context of the design and perhaps some short or long term designer's intentions, which may be difficult to anticipate. The solution to these types of problems is a major challenge of research into conceptual design support. In figure 3, these issues are collectively referred to as I3.

Once the designer has decided to reuse a particular part of a physical object, techniques for scanning and feature recognition, readily available from reverse engineering practice, can be applied to accomplish the copy-and-paste operation. However, the dedication of these techniques to the actions A1 to A5 of the proposed methodology is not without implications.

Activity A1 is the creation of a point cloud from a physical object. Standard 3D digitizing techniques can be applied, noting that for conceptual shape generation situation a low-density, low-accuracy, high-speed data acquisition suffices.

Activity A2 is the generation of a geometric model from the digitized points. Typically this involves the fitting of surfaces to points and the derivation of adjacency properties (the topology) of the surfaces constituting the model. Commercial software is available to almost automatically produce a B-rep model from unordered, though dense and accurate data points. The treatment of some types of shape features is commonly semi-automatic (Varady 1997). Also common is that these shape features are represented as a static instance but otherwise not editable.

Activity A3 is the creation of editable (freeform) shape features from the geometric model. The most common approach is based on some form of feature recognition, where instances of one of the predefined feature types are matched with the geometric model obtained after *A*2. This assumes that a sufficiently rich set of feature types is available. Another approach is the creation of a user-defined feature, where the created type is dedicated to the current design intent. This is probably the part of the methodology which is hardest to accomplish in regard to requirements 4 and 5 listed in section 2. Both the computation (discussed in the next section) and the user interaction aspects (denoted I3 in figure 3) pose serious issues.

Activity A4 is a shortcut skipping over the geometric model. In two situations it is practical to implement this shortcut. First, when the design context is spatially local, the FFF type and instance generation can be determined from data in a well-defined region of the point cloud. Then the raw measurements are sufficient to resolve feature recognition and fitting. Second, when intensive user involvement is needed (*e.g.* when the user interactively defines or searches among candidate feature types), then the geometric model as an intermediate representation becomes a too heavy computational burden.

Activity A5 is the embedding of the FFF into the current model. The requirements of this process can be largely implicit. For example, the aesthetic designer may expect that any newly inserted shape becomes tangentially continuously connected to the model. Influence from the user is, however, needed when two FFFs spatially overlap and a decision needs to be made about the dominance of one feature over others (Eversheim 2000).

Once the current CAD model has been updated, the cycle may be traversed anew. The user controls the select/copy/paste/adapt activities through the influences I1, I2 and I3. The activities A1 and A2 can be highly automated, whereas A3, A4 and A5 need some human assistance. Here appears a contrast with common applications of 3D digitizing and reverse engineering methods, where a high degree of automation is striven for. The proposed methodology for shape reuse is *intrinsically* non-automatic, as asserted by the influences I1 and I3, both representing major designer's decisions. This designer's active interest specifically provides the key to the computational feasibility of the workflow. The computer-based implementation of shape feature location, recognition and fitting which are generally not practical in a fully automated process, now become achievable if we permit the designer the control that he/she wishes.

COMPUTATIONAL FEASIBILITY

In this section we address the computational issues of the functions A3, A4 and A5. The functions A1 (object scanning) and A2 (geometric model generation from the measured points), although not fully solved problems, can be considered as being readily available in commercial packages such as Imageware and Geomagic (Meiritz 1999). Various forms of reverse engineering constitute the business drive for these systems. Of some software products it is even claimed that the conversion of a physical model into a common B-rep model can be accomplished fully automatically, provided that the data is sufficiently dense and accurate.

As mentioned, shape reuse methods do not depend on automatic solid model reconstruction, but rather on the interactive handling of selected portions of shape. With regard to the requirements listed previously we comment about the feasibility of A3, A4 and A5 subprocesses and we show some recent results from research into the implementation of the methodology.

Before going into details of the A3/4/5 activities we make two observations. First, A3 and A4 differ in only a few technical aspects; both actions result in the FFF. We will see that whether the derivation of a FFF from point data or from a geometric model is only of minor importance. Second, we should at least distinguish among two types of freeform features. One that emphasizes the identification and representation of the FFF and one that emphasizes its being variational. Most of the results in freeform feature recognition are dedicated to the first type of FFFs. They are identified within the geometric model (or within the point data) by detection of points or curves of high local curvature and subsequent bounding or separating relatively smooth surface regions. Once these regions have been detected, higher level constructs such as steps, protrusions, cavities and holes can be formed. A survey of these techniques can be found in Varady (1997). However, what we need are FFFs that can be manipulated by user or by machine, in order to support shape design. Hence, not only the shape instance should become available but also the class it belongs to and the parameter types and values for subsequent variation. This implies some sort of fitting of one or several candidate shapes against the geometric model or the point data, where the candidate shape is defined by some set of parameters. If the shape matching between data and parameterized shape is satisfactory (with regard to a still to be defined criterion) then both feature type and instance are available. Techniques for this type of feature finding can be found abundantly in the literature of feature recognition. However, most of the methods aim at the domains of 2D images and at the parameterization of affine transformations of the candidate shapes only. Variations concerning *the intrinsic shape* of the candidate entities is usually not under parameter control, as this would lead to very high dimensional search spaces.

To overcome the problem of a too high dimensionality of the search space several breakdown strategies can be applied, known from optimization techniques. Which breakdown strategy should be followed for a particular fitting problem is generally not obvious. We have tested a number of different strategies and compared them against criteria for computational load, robustness and correctness. We took the problem of locating and fitting a ridge, observable in a measured surface. The problem was formulated by quantifying the difference between the measured ridge and computer-generated shape templates of the type "ridge". Apart from the 6 D0F for the position and orientation of this template, there at least two intrinsic shape parameters characterizing the ridge, namely its height and its width, see figure 4 (Vergeest 2000).

After the fit the following results have been obtained..

1. A goodness-of-match between the hypothesized feature and the surface data. In reported work the goodness is according a measure based on a variant of the Hausdorff distance (Vergeest 2001) between the template and the surface data.
2. Values for the parameters owing to the optimal feature instance.
3. A geometric representation of the instance, which can be merged with the current model.
4. Implicitly, a parameterized feature has become available that allows the designer to deviate from the instance found, by controlling the parameters.

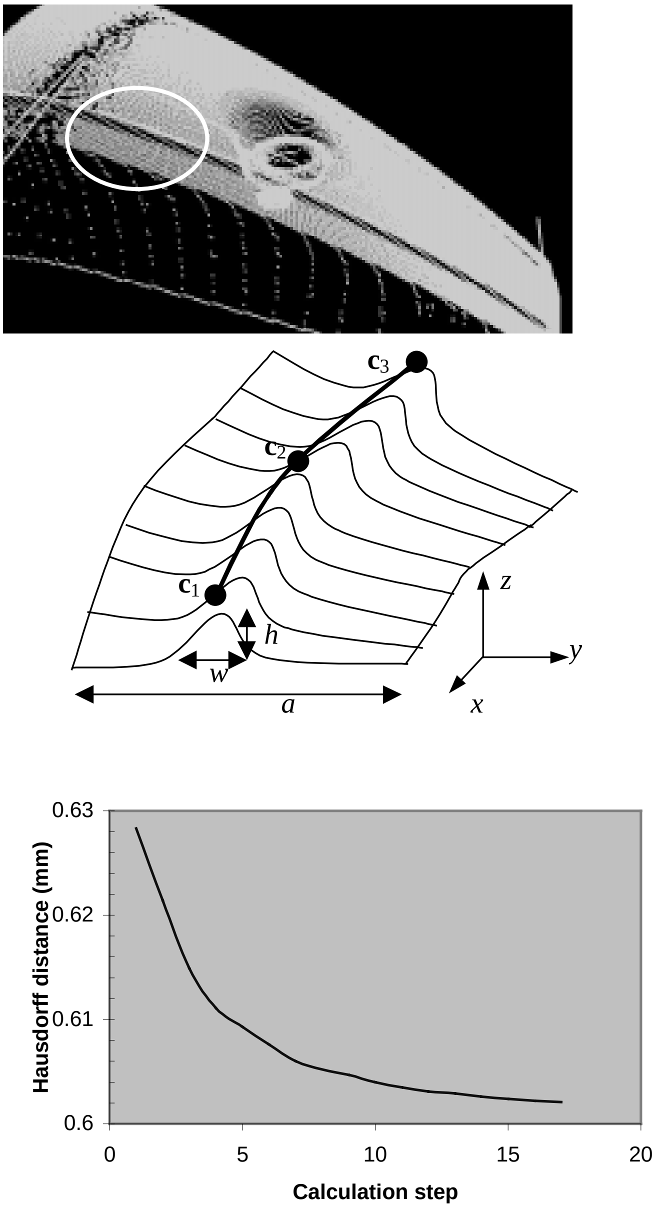

Figure 4 A low-cost desktop laser scanner was used to scan the object containing the ridge. A candidate feature shape, or template was fitted to the measured surface data; the drop of the Hausdorff distance between template and data as a function of the number of iterations is shown in the plot.

The latter result is of dominant importance for the FFF to be more than a hardwired copy of the measured physical feature. The whole fitting procedure can be accomplished in a matter of seconds if the fitting conditions are set properly (Spanjaard 2001). Initial stepping can be performed using extremely small portions of the data, and what is even more important, since the user is *intrinsically* involved in the process (especially through influence I1), the search space is naturally limited. Also the real-time control of the parameters is feasible, as was demonstrated for displacement features earlier (van Elsas 1998), see figure 5. In this work also the merge of FFFs into an existing CAD model was proven to be feasible. This indicates at least that the activity A5 can be implemented in a realistic fashion.

There seems to be no fundamental difficulty in the computation of any of the activities A1 to A5. However, the implementation should be dedicated to the application at hand, namely the (possibly very) coarse extraction of precedent shape and its insertion into the new design whilst taking advantage of the designer's involvement.

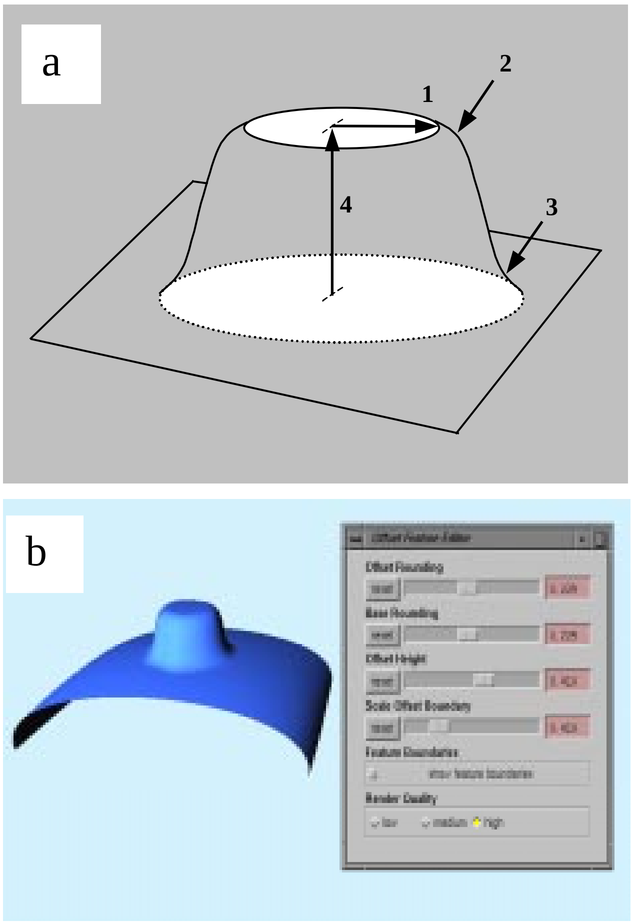

Figure 5. The four key parameters of the freeform protrusion can be interactively modified, while preserving the geometric continuity conditions.

CONCLUSIONS AND ONGOING WORK

We have obtained a preliminary evaluation of a methodology for the reuse of freeform shape. Regarding the 5 major activities, A1 and A2 (scanning and surface reconstruction) can be highly automated using available technology. The activities A3 and A4 (creation of an editable FFF from scanned points directly or from the initial geometric model, respectively) are intrinsically not fully automatic and major issues need to be resolved before these activities can be computer supported. We have presented some simple examples of A4 to indicate the potential of computer support and to clarify the research problem. We also indicated that the techniques for manipulation are very feasible and effective, once the appropriate FFF type has been created. Activity A5 is not specific to a shape reuse application and is already addressed in contemporary CAD research.

The proposed methodology can be considered as specialization of a more general methodology for design reuse. It can be envisioned that the process for shape design reuse, as depicted in figure 2, can be extended to design domains beyond shape. The technical feasibility of the shape reuse process was partially verified in this paper. However, these results do not automatically carry over to different design domains, where, for example, measures for similarity among design concepts are less obvious than those for similarity among shapes. Design reuse is typically relevant only during a limited period of the entire design process. Therefore the embedding of the shape reuse methodology into the process of product development should be explicitly addressed.

Research is ongoing into the applicability of shape reuse. We are investigating in which situations reuse could be effective and when it would, on the contrary, limit the designer's creativity. Data for this investigation are being gathered from practicing designers and engineers in industry, and from various types of experimental designs (Wiegers 2001). These findings will direct the implementation and testing of new, editable and reusable freeform features.

REFERENCES

C.K. Au, M.M.F. Yuen (2000), "A semantic feature language for sculptured object modelling". Computer-Aided Design 32, pp 63-74.

J.C. Cavendish (1995), "Integrating feature based surface design with free form deformation". Computer-Aided Design, Vol. 27, Nr. 9, pp 703-711.

A.H.B. Duffy and A.F. Ferns, "An analysis of design reuse benefits". U. Lindemann *et al.* (Eds.), proceedings of the ICED99 Conference, Technische Universität München, 1999, pp 799-804.

P.A. van Elsas and J.S.M. Vergeest (1998), "Displacement feature modelling for conceptual design". Computer-Aided Design, Vol. 30, Nr. 1, pp 19-27.

W. Eversheim, C. Deckert, M. Westekemper (2000), "Increasing efficiency through integration of freeform features into the CAD/CAM-chain. Proceedings of the 9th Symposium on Product Data Technology Europe 2000, Quality Marketing Services, Sandhurst, pp 355-362.

M. Fontana, F. Giannini and M. Meirana (1999), "A free form feature taxonomy". P. Brunet and R. Scopigno (Eds.), Proc. Eurographics'99, Computer Graphics Forum, Vol. 18, Nr. 3, pp C-107 - C-118.

N.N.Z. Gindy (1989), "A hierarchical structure for from feature". Journal of Production Research, 27, pp 2089-2103.

G. Goldschmidt (1991), "The dialectics of sketching". Creativity Research Journal, Vol. 4, Nr. 2, pp 123-143.

W.M. Hsu, J.F. Hughes, H. Kaufman (1992), "Direct manipulation of free-form deformations". ACM Computer Graphics 26(2), pp 177-184.

T. De Martino, B. Falcidieno, F. Giannini, S. Hassinger and J. Ovtcharova (1994), "Feature-based modelling by integrating design and recognition approaches". Computer-Aided Design 26(8), pp 646-652.

B. Meiritz (Ed.) (1999), Conference on Reverse Engineering, 3D scanning and a shortcut to modelling. Danish Technological Institute, Aarhus.

Mitchell, S.R., Jones, R. and Catchpole, G.(2000), "Modelling a Thin Section Sculptured Product Using Extended from Feature Methods". Journal of Engineering Design , 11(4).

B. Poldermann and I. Horváth (1996), "Surface design based on parametrized surface features". I. Horváth and Károly Váradi (Eds.), Proc. Int. Symposium on Tools and Methods for Concurrent Engineering, Institute of Machine Design, Budapest, pp 432-446.

S.N. Smyth and D.R. Wallace (2000), "Towards the synthesis of aesthetic product form". Proc. DETC2000/DTM-14554, ASME, New York.

S. Spanjaard (2001), "Comparing different fitting strategies for matching two 3D point sets usinga multivariable minimizer". Submitted to DETC2000/CIE, ASME, New York.

T. Várady, R.R. Martin and J. Cox (1997), “Reverse engineering of geometrical models - an introduction”. Computer-Aided Design, Vol. 29, No. 4, pp 255-268.

J.S.M. Vergeest, S. Spanjaard, I. Horvath, J. Jelier (2000), "Fitting freeform shape patterns to scanned 3D objects". Proc. ASME Design Engineering Technical Conf., DETC2000/CIE-14656, ASME, New York.

J.S.M. Vergeest and S. Spanjaard (2001), "The computation of 3D pattern matching as a tool for conceptual freeform shape design". Proc. ASME Design Engineering Technical Conferences, DETC2001/DAC-21139, ASME, New York.

T. Wiegers and J.S.M. Vergeest (2001), "Extraction of CAD tool requirements from industry and from experimental design projects". Proc. ASME Design Engineering Technical Conferences, DETC2001/DAC-21144, ASME, New York.

Proceedings of DETC'01
2001 ASME Design Engineering Technical Conferences
And Computers and Information in Engineering Conference
Pittsburgh, Pennsylvania, September 9-12, 2001

DETC'01/DTM-21708

A methodology for reusing freeform shape content

Joris S.M. Vergeest
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
j.s.m.vergeest@io.tudelft.nl

Imre Horváth
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
i.horváth@io.tudelft.nl

Sander Spanjaard
Delft University of Technology
Faculty of Design, Engineering and Production
Jaffalaan 9, NL-2628 CA Delft
The Netherlands
s.spanjaard@its.tudelft.nl

ABSTRACT

The reuse of precedent designs is a significant profit factor in new product development. In industry there is a tendency to enhance the reuse process by applying digital scanning of 3D parts, sampling imported, nonnative CAD models or by deploying a digital library of design concepts. The data thus obtained should be inserted into the design model. The available techniques typically originate from reverse engineering applications. However, to support shape reuse during conceptual design a dedicated methodology and workflow are needed. Using our methodology, the designer selects existing products, or parts, or portions of them. Then he/she specifies where and how the selected portion should be inserted into the new design. The key issue of the methodology is the explicit distinction between the variables that the designer does or does not wants to control. The underlying technology, including shape matching, shape parameter fitting and shape merging must be mostly invisible to the user, except for those controls that intrinsically affect the resulting shape. One application of the methodology is a freeform feature copy-and-paste facility based on 3D scanning and fitting of existent designs. The technical feasibility of such an approach will be addressed.

INTRODUCTION

In new product development two trends are becoming of dominating importance, 1) enhancement of freeform shape design and its accomplishment using *freeform features,* 2) the integration of virtual and physical modeling in the freeform domain, which implicitly calls for *shape reuse* methods.

Freeform features

Especially for consumer products the outer appearance represents a key value of the design. Despite the considerable proliferation of tools for computer-aided industrial design (CAID), the creation, manipulation and management of freeform objects is still only marginally supported in comparison to the impact exerted by *e.g.* mechanical CAD systems. Nevertheless, the expectations regarding the product's aesthetic and ergonomic quality are very high, and design proposals are to be delivered under severe time pressure. This led to a number of attempts to apply principles of feature-based design from mechanical design in the domain of freeform shape modeling and styling design. Recently, (Au 2000) proposed a grammatical feature modeling to aggregate non-regular shaped features into a sculptured object. The features and their relations to the object and to other features formed the vocabulary of a feature language. The language was partially object-specific and explicitly contained the hierarchical structure of the design and rules for connecting spatially adjacent features. The final geometry of the individual feature instances was derived by solving the constraints implied by the rules. The system proposed by Au *et al.* enables the designer to quickly generate object variations within a given framework of the object's structure. The object specificity, however, appears as a limitation of the feature approach to freeform object modeling. On the other hand it has been shown in (Mitchell 2000) that an object-specific feature anatomy enables accurate performance predictions of the product design.

A big contrast between regular-shaped features and freeform features is that the latter can hardly be predefined generically, but should evolve in a specific design context and be customized consequently. This seems to call for a different design workflow than the one known for mechanical feature-based design. Especially the principle of aggregation of feature instances as to form the entire design model seems very restrictive. This may explain the considerable attention that has been payed to the development of detail features modeling techniques, as for example by (Cavendish 1995) and (van Elsas 1998). Also the proposal for the classification of freeform features by (Fontana 1999) emphasizes the detailed features over the structural features. In (Poldermann 1995), a distinction is made between global and local surface features. Alternative freeform feature classifications have been proposed by (Gindy 1989) and (Eversheim 2000), both aiming at the incorporation of manufacturing aspects into the designed shape.

Another distinction between regular-shapes and freeform features is their spatial boundary, which is very clear for steps, holes, slots and the like but sometimes quite ambiguous for styling lines, smooth cavities etc. Perhaps, surface features should be treated in a dual way: both as a set of shape handles (Hsu 1992) and as constituents of an object.

Shape reuse

Shape reuse is the application of precedent modeling effort in the current design model. It is a specific kind of design reuse, which is generally appreciated as a profit factor in new product design (Duffy 1999). The most commonly known form of shape reuse is the usage of digital part libraries for CAD. Especially when the "standard parts" are parameterized models, a significant reduction of design effort is achieved (De Martino 1994). However, the success is limited to frequently recurring types of shapes, which are typically regular shaped. It is therefore challenging to enable the reuse of shapes beyond the domain of regular shapes. To achieve this, at least three key issues need to be resolved: 1) the creation of new freeform shape types (*i.e.* features) at runtime, as opposed to the formation of a hardwired library of standard parts, 2) finding a parameterization scheme for freeform features, 3) the development and verification of an efficient workflow incorporating design by reuse. It depends on this latter issue how to approach the problem of user-defined freeform features and how to parameterize them. The requirements from the end-user (*i.e.* from the designer or stylist using the envisaged system) determine which techniques known from mechanical feature design can be adopted and which cannot.
Those requirements are addressed in the following section. From these requirements we derive a methodology and workflow for design by reuse in the freeform shape domain. Then, we review the key elements of the methodology against their computational feasibility and we present some examples of solved subproblems.

.

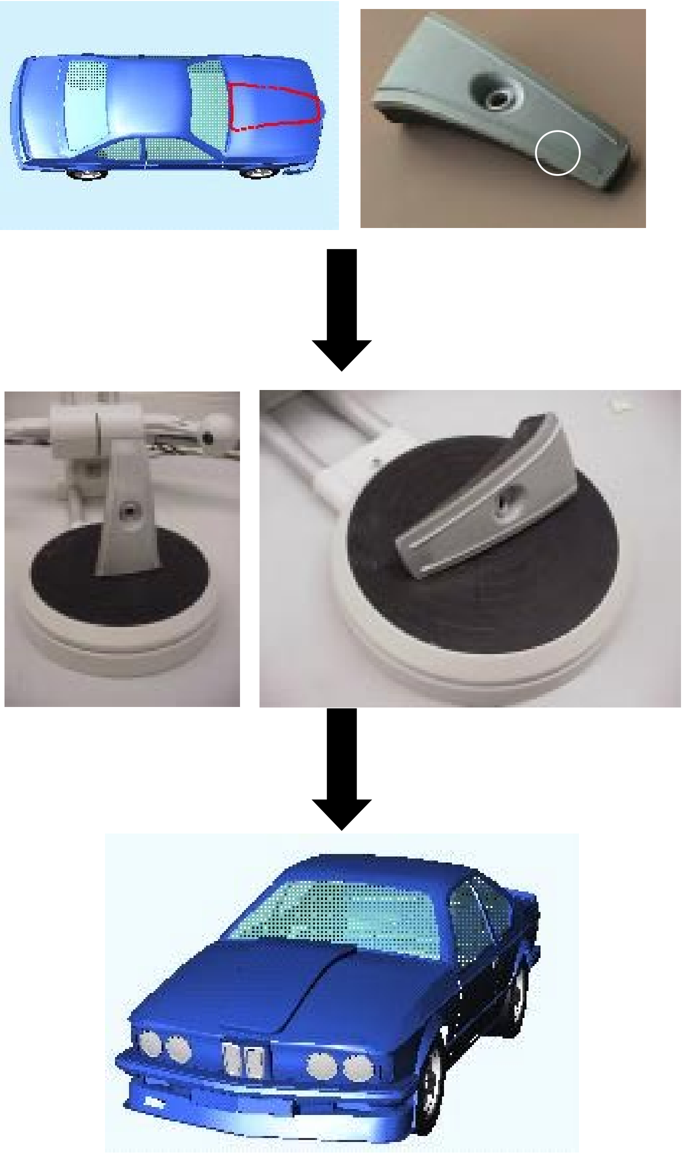

Figure 1. Simplified depiction of the workflow of shape reuse. Top: the designer intends to include a ridge to the car motor hood, where the ridge should be similar to the one observed in the object (see circle). The ridge is scanned using a desktop laser scanner and the ridge is then inserted into the CAD model of the car.

REQUIREMENTS FOR SHAPE REUSE IN CONCEPTUAL DESIGN

The scope of the methodology is the explicit insertion of an encountered shape feature into the current design, a simple example is depicted in figure 1. Possible sources of the features include physical objects (which then need to be scanned) or portions of 3D digital models, where a model can be fully nonnative (*e.g.* found on the Internet) or it can be the current design itself, to mention two extremes

The principle functions that should be provided according to the methodology are:

1. A selection method that enables the user to designate existing shapes or portions of them. The existing shapes can be either boundaries of physical objects, or shape models. Shape models include physical 2D sketches, but most commonly they will be available in the form of computer representations. The selected portions are intended for reuse.
2. A conversion method that creates a computer-based representation of the selected shape. In case the selected shape is encountered on a physical object, a process of 3D digitizing and surface reconstruction will be involved. In case the selected shape is already available in digital form, then only the surface reconstruction is needed. Only incidentally, the original surface representation will be adequate for the purpose of reusing the shape in the destination model. Therefore, generally, a representation conversion will be needed.
3. A selection method that enables the user to describe the shape-to-be-reused as an entity that can be manipulated. Only incidentally the selected shape has precisely the geometry as required in the destination model, even when (due to requirement 2) the internal representation is of the right type. Generally, some modifications of the shape are needed before it is inserted into the destination shape. In the case of freeform features, these modifications are typically applied by using parameters. The selection comprises the specification of a type of shape, or the feature type, that should be applied to the selected shape. These types can be either predefined or be user-defined.
4. The internal representation of the selected shape should be upgraded to the representation implied by the chosen feature type. This involves the matching of the shape (either its available representation, or the originally measured points) to synthetic shapes of the given feature type. If a match is found then the selected shape is available as an instance of the feature type, and hence ready for the user to be controlled by its defining parameters.
5. The user must be able to revise his/her choice of feature type. If during the reuse process it appears that a particular type of manipulation is not supported by the system, *i.e.* was not anticipated when the type of feature, and hence its parameterization, was selected, then either the new parameter should be included in the set of parameters, or the feature type should be revised. This involves the technique mentioned in requirement 4.
6. The user must be able to evaluate the result of insertion of

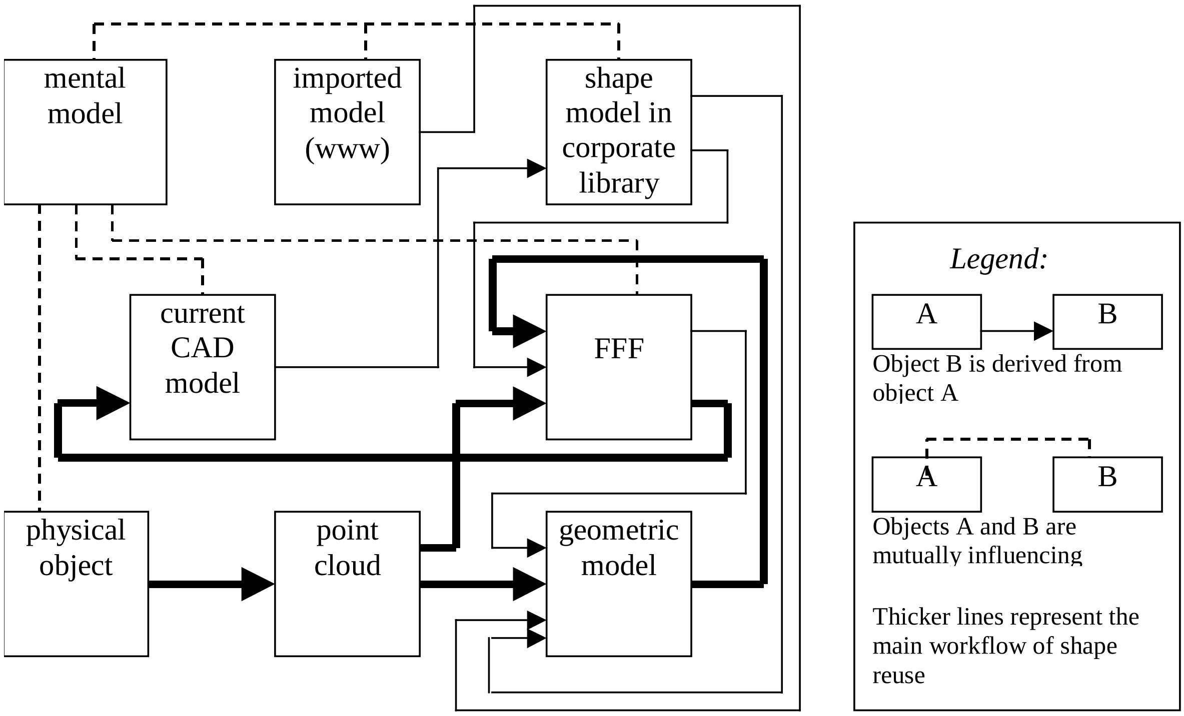

Figure 2. The eight principle objects relevant in the shape reuse methodology. The lines and arrows represent dependencies among the objects.

the shape feature into the destination shape instantly during his/her controlling the feature parameters. This is to adequately merge the feature with the new design. Placement and scaling control are extremely important, as will be particular shape manipulations, permitted by the feature definition.

7. There should be automatic fine tuning for merge of the feature with the destination shape so that the transition between the two entities conforms to smoothness criteria. These criteria are usually defaulted to geometric continuity conditions, but they may be user-defined as well.
8. A facility should be provided to store all user-defined feature types for later reference. In this way the system expands its library of available shape parameterizations.

Requirement 5 addresses a key challenge of contemporary CAD research. If a designer changes his/her mind about a once decided type of shape parameterization, then it is highly untrivial to actually re-represent the shape. Yet, the flexibility to switch among parameterizations would enormously increase the computer-supported design performance.

DESCRIPTION OF THE METHODOLOGY

To describe the shape reuse methodology complying to the above requirements we need to account for the following objects depicted in figure 1.

Current CAD model. This is the presumed design model at a particular stage of development. It is typically a computer-based model in one of the representation known for CAD system, *e.g.* a boundary representation and/or a model built of freeform surfaces. The precise type of representation is of no relevance to the methodology (of course it will be to its technical implementation, as discussed in the next section). The current CAD model is understood as the model of most relevance to the designer (or design team). All other objects can be regarded as auxiliary to the current CAD model (although in a particular implementation some of the models may be available on one system, or even be integrated).

Physical object. This can be any observable thing. Typically it is a manmade product, for example a desk light, a telephone set, a car, or a clay model. If the object is tangible then it can be digitized by a 3D scanning system. If the physical object contains a portion having a shape that could be used in the next step of development of the current CAD model, then the physical object is relevant in the shape reuse methodology.

Point cloud. The result of object scanning is a point cloud, a set of measured locations on the surface of the physical object. Depending on the applied scanning technology, the points may be implicitly ordered into grid structures or the like, but they can be totally unordered as well.

Geometric model. The geometric model can be constructed from a point cloud, as is common practice in reverse engineering applications (Várady 1997). Typically, the geometric model is represented by a set of surfaces (with or without connectivity information) or as a B-rep. It can be considered subservient to the current CAD model (defined above). The geometric model may be a temporary representation of a portion of a shape, which can later be inserted into the current CAD model. In some implementations the geometric model and the current CAD model are identical.

Freeform Feature (FFF). This is a shape specified by a (multidimensional) parameter value; an instance generated from a feature type and the parameter value.

Imported model. Any shape model retrieved from the internet or sent by a partner in the design process. Possibly, the representation of the imported model is different from any of the representation forms known by the receiving CAD system.

Shape model in corporate library. Corporate library represents a repository of shape models which is local to the user or the company. A common implementation form is a company-specific parts library, but it can also take the form of a database private to a CAD system's user. The shape models in the library can be any of the types *point cloud, geometric model, imported model, current CAD model* or *FFF*.

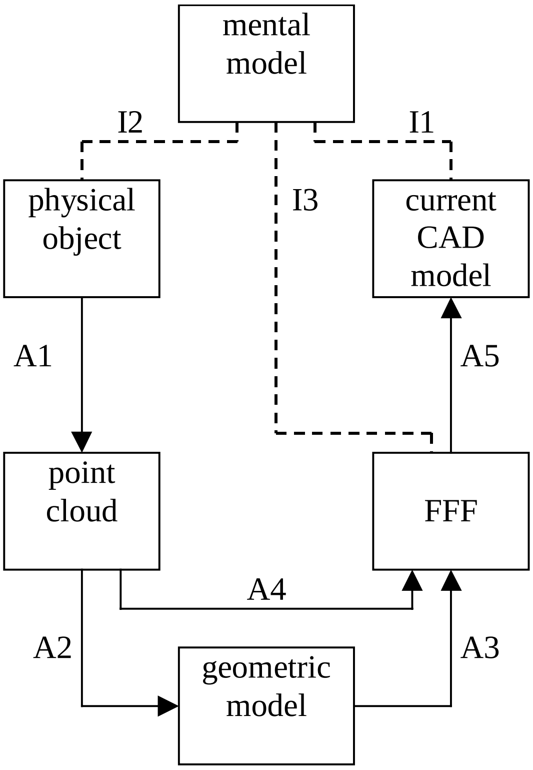

Figure 3. Basic cycle of the shape reuse process. For the legend we refer to Figure 2.

Mental model. This is an identifier for the interface between the human designer and the otherwise computer-based objects. The mental model is then a presupposed image or understanding that a subject has about the designed shape. We will assume particular forms of interplay between the mental model and the exteriorly presented shape (Goldschmidt 1991).

Using the eight entities just defined, the workflow of shape reuse can be described on various levels of detail. On a coarse level only the thick arrows in figure 2 are relevant, and the design process shown in figure 3 is implied. This picture is an oversimplification; it emphasizes the actual information flow inherent to the shape reuse process. As mentioned, physical object, for example, can be replaced by imported model, without invalidating the picture.

The user interacts with three objects, the current CAD model, the physical object and the FFF, indicated in figure 3 by the influences I1, I2 and I3, respectively. The object types have been defined above. I1 is an identifier of what is commonly understood by computer-aided design in any of its forms. In this paper we assume that a user interactively manipulates the current CAD model, part of which represents the geometric shape of the design. On different levels of abstraction the user may wish to reuse precedent designs or shapes. It can be on an informal level, where the designer applies methods or actions that he/she recalls from past experience or learnt strategies. On a more concrete level, the designer may want to, or is requested to, apply particular archetypes or patterns into the new design, for example to comply with product's brand styling guidelines (Smyth 2000). On this level, influence I2 enters the methodology when the user refers to a physical model to obtain a starting point for form evolution, although the starting form can also be created explicitly (using I1) or by any other input means. In Smyth's approach shape concept generation starts from a skeleton archetype which then evolves and mutates through some parameter space; the user selects the most appropriate instance. When shape reuse has a spatially local context, then the embedding of the new shape feature into the current design is crucial (this is activity A5 addressed later). This embedding may be partially automated but in general will involve some attention of the designer; this influence is denoted I3. In an even more concrete manner there are at least two other possible scenarios, 1) the designer encounters a physical (or digital) object containing a shape feature that he/she believes is useful in the current design, or 2) the designer searches among physical objects with the aim to borrow some or all of its shape for inclusion into the current design. These scenarios involve human search, recognition and judgement, in figure 3 modeled by the influences I1, I2 and I3.

A fundamental factor in shape reuse is the adaptation of the reused feature in the current design. We already mentioned the geometric embedding of the feature, but also its shape might need significant modification, which should be supported by a dedicated parameterization of the feature. Which parameters need to be regulated depends on the context of the design and perhaps some short or long term designer's intentions, which may be difficult to anticipate. The solution to these types of problems is a major challenge of research into conceptual design support. In figure 3, these issues are collectively referred to as I3.

Once the designer has decided to reuse a particular part of a physical object, techniques for scanning and feature recognition, readily available from reverse engineering practice, can be applied to accomplish the copy-and-paste operation. However, the dedication of these techniques to the actions A1 to A5 of the proposed methodology is not without implications.

Activity A1 is the creation of a point cloud from a physical object. Standard 3D digitizing techniques can be applied, noting that for conceptual shape generation situation a low-density, low-accuracy, high-speed data acquisition suffices.

Activity A2 is the generation of a geometric model from the digitized points. Typically this involves the fitting of surfaces to points and the derivation of adjacency properties (the topology) of the surfaces constituting the model. Commercial software is available to almost automatically produce a B-rep model from unordered, though dense and accurate data points. The treatment of some types of shape features is commonly semi-automatic (Varady 1997). Also common is that these shape features are represented as a static instance but otherwise not editable.

Activity A3 is the creation of editable (freeform) shape features from the geometric model. The most common approach is based on some form of feature recognition, where instances of one of the predefined feature types are matched with the geometric model obtained after *A*2. This assumes that a sufficiently rich set of feature types is available. Another approach is the creation of a user-defined feature, where the created type is dedicated to the current design intent. This is probably the part of the methodology which is hardest to accomplish in regard to requirements 4 and 5 listed in section 2. Both the computation (discussed in the next section) and the user interaction aspects (denoted I3 in figure 3) pose serious issues.

Activity A4 is a shortcut skipping over the geometric model. In two situations it is practical to implement this shortcut. First, when the design context is spatially local, the FFF type and instance generation can be determined from data in a well-defined region of the point cloud. Then the raw measurements are sufficient to resolve feature recognition and fitting. Second, when intensive user involvement is needed (*e.g.* when the user interactively defines or searches among candidate feature types), then the geometric model as an intermediate representation becomes a too heavy computational burden.

Activity A5 is the embedding of the FFF into the current model. The requirements of this process can be largely implicit. For example, the aesthetic designer may expect that any newly inserted shape becomes tangentially continuously connected to the model. Influence from the user is, however, needed when two FFFs spatially overlap and a decision needs to be made about the dominance of one feature over others (Eversheim 2000).

Once the current CAD model has been updated, the cycle may be traversed anew. The user controls the select/copy/paste/adapt activities through the influences I1, I2 and I3. The activities A1 and A2 can be highly automated, whereas A3, A4 and A5 need some human assistance. Here appears a contrast with common applications of 3D digitizing and reverse engineering methods, where a high degree of automation is striven for. The proposed methodology for shape reuse is *intrinsically* non-automatic, as asserted by the influences I1 and I3, both representing major designer's decisions. This designer's active interest specifically provides the key to the computational feasibility of the workflow. The computer-based implementation of shape feature location, recognition and fitting which are generally not practical in a fully automated process, now become achievable if we permit the designer the control that he/she wishes.

COMPUTATIONAL FEASIBILITY

In this section we address the computational issues of the functions A3, A4 and A5. The functions A1 (object scanning) and A2 (geometric model generation from the measured points), although not fully solved problems, can be considered as being readily available in commercial packages such as Imageware and Geomagic (Meiritz 1999). Various forms of reverse engineering constitute the business drive for these systems. Of some software products it is even claimed that the conversion of a physical model into a common B-rep model can be accomplished fully automatically, provided that the data is sufficiently dense and accurate.

As mentioned, shape reuse methods do not depend on automatic solid model reconstruction, but rather on the interactive handling of selected portions of shape. With regard to the requirements listed previously we comment about the feasibility of A3, A4 and A5 subprocesses and we show some recent results from research into the implementation of the methodology.

Before going into details of the A3/4/5 activities we make two observations. First, A3 and A4 differ in only a few technical aspects; both actions result in the FFF. We will see that whether the derivation of a FFF from point data or from a geometric model is only of minor importance. Second, we should at least distinguish among two types of freeform features. One that emphasizes the identification and representation of the FFF and one that emphasizes its being variational. Most of the results in freeform feature recognition are dedicated to the first type of FFFs. They are identified within the geometric model (or within the point data) by detection of points or curves of high local curvature and subsequent bounding or separating relatively smooth surface regions. Once these regions have been detected, higher level constructs such as steps, protrusions, cavities and holes can be formed. A survey of these techniques can be found in Varady (1997). However, what we need are FFFs that can be manipulated by user or by machine, in order to support shape design. Hence, not only the shape instance should become available but also the class it belongs to and the parameter types and values for subsequent variation. This implies some sort of fitting of one or several candidate shapes against the geometric model or the point data, where the candidate shape is defined by some set of parameters. If the shape matching between data and parameterized shape is satisfactory (with regard to a still to be defined criterion) then both feature type and instance are available. Techniques for this type of feature finding can be found abundantly in the literature of feature recognition. However, most of the methods aim at the domains of 2D images and at the parameterization of affine transformations of the candidate shapes only. Variations concerning *the intrinsic shape* of the candidate entities is usually not under parameter control, as this would lead to very high dimensional search spaces.

To overcome the problem of a too high dimensionality of the search space several breakdown strategies can be applied, known from optimization techniques. Which breakdown strategy should be followed for a particular fitting problem is generally not obvious. We have tested a number of different strategies and compared them against criteria for computational load, robustness and correctness. We took the problem of locating and fitting a ridge, observable in a measured surface. The problem was formulated by quantifying the difference between the measured ridge and computer-generated shape templates of the type "ridge". Apart from the 6 D0F for the position and orientation of this template, there at least two intrinsic shape parameters characterizing the ridge, namely its height and its width, see figure 4 (Vergeest 2000).

After the fit the following results have been obtained..

1. A goodness-of-match between the hypothesized feature and the surface data. In reported work the goodness is according a measure based on a variant of the Hausdorff distance (Vergeest 2001) between the template and the surface data.
2. Values for the parameters owing to the optimal feature instance.
3. A geometric representation of the instance, which can be merged with the current model.
4. Implicitly, a parameterized feature has become available that allows the designer to deviate from the instance found, by controlling the parameters.

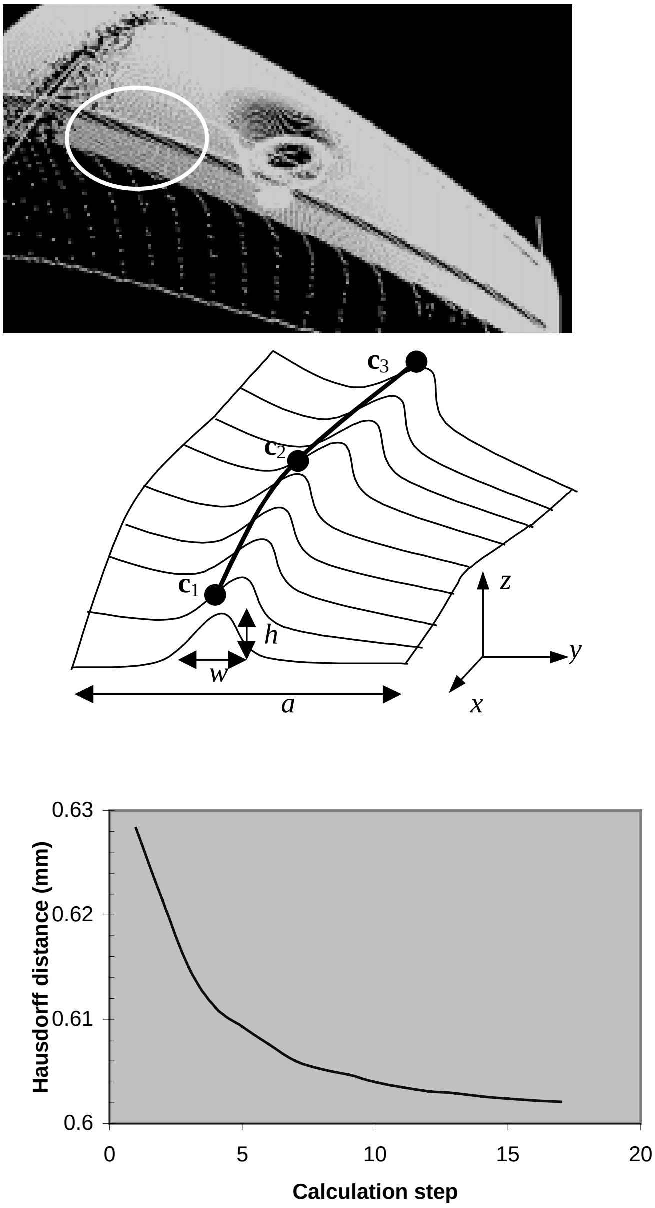

Figure 4 A low-cost desktop laser scanner was used to scan the object containing the ridge. A candidate feature shape, or template was fitted to the measured surface data; the drop of the Hausdorff distance between template and data as a function of the number of iterations is shown in the plot.

The latter result is of dominant importance for the FFF to be more than a hardwired copy of the measured physical feature. The whole fitting procedure can be accomplished in a matter of seconds if the fitting conditions are set properly (Spanjaard 2001). Initial stepping can be performed using extremely small portions of the data, and what is even more important, since the user is *intrinsically* involved in the process (especially through influence I1), the search space is naturally limited. Also the real-time control of the parameters is feasible, as was demonstrated for displacement features earlier (van Elsas 1998), see figure 5. In this work also the merge of FFFs into an existing CAD model was proven to be feasible. This indicates at least that the activity A5 can be implemented in a realistic fashion.

There seems to be no fundamental difficulty in the computation of any of the activities A1 to A5. However, the implementation should be dedicated to the application at hand, namely the (possibly very) coarse extraction of precedent shape and its insertion into the new design whilst taking advantage of the designer's involvement.

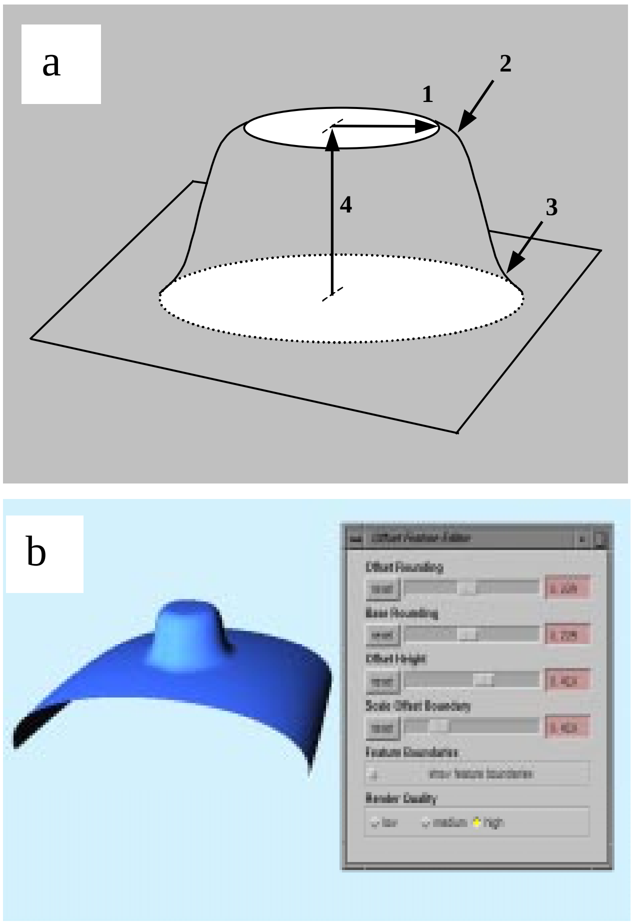

Figure 5. The four key parameters of the freeform protrusion can be interactively modified, while preserving the geometric continuity conditions.

CONCLUSIONS AND ONGOING WORK

We have obtained a preliminary evaluation of a methodology for the reuse of freeform shape. Regarding the 5 major activities, A1 and A2 (scanning and surface reconstruction) can be highly automated using available technology. The activities A3 and A4 (creation of an editable FFF from scanned points directly or from the initial geometric model, respectively) are intrinsically not fully automatic and major issues need to be resolved before these activities can be computer supported. We have presented some simple examples of A4 to indicate the potential of computer support and to clarify the research problem. We also indicated that the techniques for manipulation are very feasible and effective, once the appropriate FFF type has been created. Activity A5 is not specific to a shape reuse application and is already addressed in contemporary CAD research.

The proposed methodology can be considered as specialization of a more general methodology for design reuse. It can be envisioned that the process for shape design reuse, as depicted in figure 2, can be extended to design domains beyond shape. The technical feasibility of the shape reuse process was partially verified in this paper. However, these results do not automatically carry over to different design domains, where, for example, measures for similarity among design concepts are less obvious than those for similarity among shapes. Design reuse is typically relevant only during a limited period of the entire design process. Therefore the embedding of the shape reuse methodology into the process of product development should be explicitly addressed.

Research is ongoing into the applicability of shape reuse. We are investigating in which situations reuse could be effective and when it would, on the contrary, limit the designer's creativity. Data for this investigation are being gathered from practicing designers and engineers in industry, and from various types of experimental designs (Wiegers 2001). These findings will direct the implementation and testing of new, editable and reusable freeform features.

REFERENCES

C.K. Au, M.M.F. Yuen (2000), "A semantic feature language for sculptured object modelling". Computer-Aided Design 32, pp 63-74.

J.C. Cavendish (1995), "Integrating feature based surface design with free form deformation". Computer-Aided Design, Vol. 27, Nr. 9, pp 703-711.

A.H.B. Duffy and A.F. Ferns, "An analysis of design reuse benefits". U. Lindemann *et al.* (Eds.), proceedings of the ICED99 Conference, Technische Universität München, 1999, pp 799-804.

P.A. van Elsas and J.S.M. Vergeest (1998), "Displacement feature modelling for conceptual design". Computer-Aided Design, Vol. 30, Nr. 1, pp 19-27.

W. Eversheim, C. Deckert, M. Westekemper (2000), "Increasing efficiency through integration of freeform features into the CAD/CAM-chain. Proceedings of the 9th Symposium on Product Data Technology Europe 2000, Quality Marketing Services, Sandhurst, pp 355-362.

M. Fontana, F. Giannini and M. Meirana (1999), "A free form feature taxonomy". P. Brunet and R. Scopigno (Eds.), Proc. Eurographics'99, Computer Graphics Forum, Vol. 18, Nr. 3, pp C-107 - C-118.

N.N.Z. Gindy (1989), "A hierarchical structure for from feature". Journal of Production Research, 27, pp 2089-2103.

G. Goldschmidt (1991), "The dialectics of sketching". Creativity Research Journal, Vol. 4, Nr. 2, pp 123-143.

W.M. Hsu, J.F. Hughes, H. Kaufman (1992), "Direct manipulation of free-form deformations". ACM Computer Graphics 26(2), pp 177-184.

T. De Martino, B. Falcidieno, F. Giannini, S. Hassinger and J. Ovtcharova (1994), "Feature-based modelling by integrating design and recognition approaches". Computer-Aided Design 26(8), pp 646-652.

B. Meiritz (Ed.) (1999), Conference on Reverse Engineering, 3D scanning and a shortcut to modelling. Danish Technological Institute, Aarhus.

Mitchell, S.R., Jones, R. and Catchpole, G.(2000), "Modelling a Thin Section Sculptured Product Using Extended from Feature Methods". Journal of Engineering Design , 11(4).

B. Poldermann and I. Horváth (1996), "Surface design based on parametrized surface features". I. Horváth and Károly Váradi (Eds.), Proc. Int. Symposium on Tools and Methods for Concurrent Engineering, Institute of Machine Design, Budapest, pp 432-446.

S.N. Smyth and D.R. Wallace (2000), "Towards the synthesis of aesthetic product form". Proc. DETC2000/DTM-14554, ASME, New York.

S. Spanjaard (2001), "Comparing different fitting strategies for matching two 3D point sets usinga multivariable minimizer". Submitted to DETC2000/CIE, ASME, New York.

T. Várady, R.R. Martin and J. Cox (1997), “Reverse engineering of geometrical models - an introduction”. Computer-Aided Design, Vol. 29, No. 4, pp 255-268.

J.S.M. Vergeest, S. Spanjaard, I. Horvath, J. Jelier (2000), "Fitting freeform shape patterns to scanned 3D objects". Proc. ASME Design Engineering Technical Conf., DETC2000/CIE-14656, ASME, New York.

J.S.M. Vergeest and S. Spanjaard (2001), "The computation of 3D pattern matching as a tool for conceptual freeform shape design". Proc. ASME Design Engineering Technical Conferences, DETC2001/DAC-21139, ASME, New York.

T. Wiegers and J.S.M. Vergeest (2001), "Extraction of CAD tool requirements from industry and from experimental design projects". Proc. ASME Design Engineering Technical Conferences, DETC2001/DAC-21144, ASME, New York.