Journal of Engineering Design, Vol. X, No. X, Month 2007, xxx-xxx

Article Type: Research Note

Semantic-based operators to support car sketching

V. CHEUTET*†‡, C.E. CATALANO‡, F. GIANNINI‡, M. MONTI‡, B. FALCIDIENO‡ and J.C. LEON†

†Laboratory Soils, Solids and Structures, Integrated Design Project, Grenoble, France
‡Istituto di Matematica Applicata e Tecnologie Informatiche – CNR, Department of Genova, Italy

*Corresponding author. Email: vincent.cheutet@hmg.inpg.fr

Abstract

This paper presents a first step in the elaboration of a semantic-based modelling environment addressed to the conceptual design phase of an industrial product. Up to now shape generation and manipulation in aesthetic design are still based on low-level geometric tools that come from the classical CAD paradigm. On the other hand, the design of a product is mainly driven by designers' creativity and high-level constraints. To capture and structure the semantics embedded in the first sketches representing the product, an ontology has been devised to guide more easily the generation and manipulation of curves that are basic elements of the product description in the early design phase. This ontology includes a taxonomy of the aesthetic curves in the automotive field and a curve manipulation setting based on a shape grammar, creating explicit connection between the two contexts.

Keywords: Aesthetic design, semantic-based modelling, shape grammar, 2D sketches, ontology

AMS Subject Classification: 62P30, 68T30

1 Introduction

Nowadays product development process is highly supported in all the various phases by computer-aided tools. Time pressure, economical constraints and the increasing distribution of the various activities among different departments and companies enforce the need of a better integration among tools and knowledge sharing. Maintaining all the knowledge related to the product and created within the different phases is crucial to avoid the invalidation of the original intent and objectives because of missing information or misunderstanding.
The employed computer-aided tools are more and more sophisticated in terms of supported functionalities, but still provide only partial solutions to this problem. From the one hand, the adoption of Product Data Management (PDM) systems provides an organisation of the product data and models, facilitating data retrieval; from the other hand, the availability of powerful and flexible knowledge technologies has brought big benefits to the Computer-Aided Design (CAD) paradigm. They allow for the development of product modelling functionalities able to incorporate prior knowledge into the product model, thus integrating all the information specific to given product categories or produced throughout different design phases. Currently available knowledge-based systems focus mainly on the functional elements of the design and do not support the management of the aesthetic knowledge; as a consequence, the industrial design process does not fully benefit of the technological progresses, being the divergence between stylist's conceptual model and the currently adopted modelling paradigm still big (Piegl 2005).

The reason is the complexity of styling process that is intrinsically a creative activity in which innovative shape solutions are characterised by fuzzy, uncertain and subjective aspects. First, the aesthetics of a product is a decisive criterion in the choices of customers; as a consequence, stylists mainly focus on the creation of appealing shapes without taking care of the details. This gives rise to complex shapes which require high level modelling tools supporting also the shape adaptation to the designer's idea.
In addition, it is very frequent that changes are required on the digital model derived from the sketch since stylists usually prefer to use pen and paper where they can easily emphasise some shape aspects that cannot be effectively realised in the real product. Also in the consecutive engineering phases, new production or functional constraints may require shape modifications. In this perspective, the big advantage of having a computer processable formalisation of the styling intent is clear, since it makes possible the definition of modelling tools able to act directly on it. On the other hand, it is not an easy task. In fact, it implies to identify and structure the elements affecting the product aesthetics, which are implicitly expressed within the sketches.
In this paper, we tackle this problem through the use of ontology, which is well recognised as an effective mean for sharing knowledge. The objective of the proposed ontology is to provide a structured representation of the semantics embedded in the product sketches and to express curve manipulation process in terms of shape characteristics and operators easily understandable by stylists. The domain knowledge represented in the ontology has been captured and validated through a deep analysis of marketing documentation and discussions held with stylists of European car builders, such as Pininfarina, BMW and SAAB (Poitou 2002), along six project years (FIORES, FIORES II). We face the problem at two levels. At the first, we identify the conceptual elements which define the structure of the visual appearance of the product, i.e. those recurrent elements that are primarily modified in order to change the product aesthetics. We focus on the automotive product category, which is a highly complex product but with very strong engineering constraints that limit the freedom in the shape. Car design has an evolutionary nature, i.e. each solution can be derived by previous ones in a complex way. This strongly differs from other styling products, in which most of the time stylists look for completely new and surprising shapes, as for example in the case of Alessi's products (Alessi).
The aesthetic key elements identified are mainly 2D curves. This can be perceived as a limit at the first glance, since the shape stylists have in mind is essentially 3D, but when looking at how they work and how they abstract the shape, we see that they essentially concentrate on specific curves e.g. profiles, sections and reflection lines, which are normally judged in a planar view (paper or CAD screen).
At the second level of our formalisation, we focus on the specification of aesthetically meaningful operators expressed through a specific shape grammar acting on a neutral curve description. Shape grammars provide a description of a shape through a concise and repeatable language, which acts only on the shape intrinsic characteristics, thus it is independent of the specific system. Moreover, comparisons between shapes can be made through their grammatical characterisations. The shape grammar introduced in the following will be used to describe shapes and manipulate them internally to the system; it will be hidden to designers, who will oppositely deal only with concepts they are familiar with. It can be considered as a neutral language between the user's knowledge and the geometric description of the shape needed by the CAS (Computer Aided Styling) or CAD system.
The paper is organised as follows; In Section 2, literature related to the different methodologies utilised has been outlined; in Section 3, a general overview on the structure of the ontology has been given, while in Section 4 and 5 the in depth description of the aesthetic and the geometric fragments follow. Finally, Section 6 deals with the connections between the two domains, and Section 7 concludes the paper.

2 Related works

Several studies aiming at identifying and exploiting aesthetically relevant product elements for improving the design process have been carried out from different perspectives. McCormack and Cagan (2004) define shape

grammars to translate the key elements of Buick vehicles into a repeatable language, which can be used to generate products consistent with the Buick brand. De Luca et al. (2005) formalise primitives and features of ancient architectural styles based on an analysis of architectural treaties. Hsiao and Wang (1998) presented a method for modifying the rough model of a car in accordance with a target character. Their approach is based mainly on the collection of customer verbal descriptions and only relies on the car proportions such as height and tail length. Cai et al. (2003) propose the idea of driving the design process by semantics words, describing shape and colour aesthetic rules. Fujita et al. (1999) proposed a methodology for designing products with integrated consideration of all aspects by introducing aesthetic features for interpretation of aesthetics and combining constraint management in geometric modelling for engineering design. Case and Karim (2005) propose to combine formal aesthetic and functional elements to be used as basic elements for genetic algorithms for shape evolution. Miura (2006) proposes an equation of aesthetic curves to be used as standard to generate, evaluate and deform curves of industrial product models. Yanagisawa and Fukuda (2004) defined a computer-based system for the appraisal communication of the customer to the provided model. Through repeated processes of shape proposal and score attribution the system should be able to provide a model that best fits with the customer preferences. Wielinga et al. (2001) develop a content-based image retrieval system for images of artefacts, which is based on an ontology of art styles derived from the Art and Architecture Thesaurus (AAT). Although all these research activities represent meaningful steps towards a semantic-based environment for industrial design, the problem of describing shape characteristics from an aesthetic perspective has not been properly addressed yet in a form suited to take into account and express the shape transformations occurring during the various design stages.

First of all to create such a semantic-based environment for aesthetic design, stylists' way of working has to be taken into account, especially because the automotive sector has the advantage of a more structured pipeline in the creation phase. Ontology technology seems to be a valid framework for structuring such knowledge. In fact, an ontology is a "specification of a conceptualisation" (Gruber 1993), i.e. a system which describes concepts and the relationships between them. Ontologies have been employed in the artificial intelligence community to describe a variety of domains, becoming a fundamental technological mechanism for sharing, reusing and analysing information.

In the last years, several works exploited ontology capabilities to integrate knowledge at different stages of product design providing a semantic-based environment for the design process (Kopena and Regli 2003, Brunetti and Grimm 2005, Kitamura and Mizoguchi 2004, van der Vegte et al. 2004). Patil et al. (2005) propose an ontological approach to formalise product semantics into a product semantic representation language for addressing the product life-cycle management. In Mizoguchi et al. (2000) and Posada et al. (2005), ontology is used in the context of oil–refinery plant; in the first paper, the objective is to formalise both the application domain and the tasks representing groups of activities in the plant. In the second, an ontology has been devised for a semantics-driven simplification of CAD models, applied to the visualisation and the design review of large plant models.

Finally, in our work for a powerful formalisation of curves and operators independent of the geometric description chosen by the CAD system, a special shape grammar has been customised. In fact, a shape grammar (Stiny 1980) defines a set of precise generating rules, which, in turn, can be used to produce a language of shapes to generate infinitely many instances of shape arrangements; the shape vocabulary constitutes the knowledge-base of an expert system that would create new design alternatives. Shape grammars have been extensively used in well-defined design domain, as architecture (Cagdas 1997), but also in product design (Hsiao and Chan 1997, Smyth and Edmonds 2000, McCormack and Cagan 2004). But all these shape grammars are either too much context-dependent to be applied to other contexts or limited to relatively regular forms and do not support readily aesthetic design. Moreover, the exploitation of shape

grammars has been slow, partly due to the lack of good interaction between the user and the system (Chase 2002).
Among all the shape grammars, the one developed by Leyton (1988), dealing with 2D C^2 curves seems the most appropriate for our context and we already adopted it for feature manipulation (Pernot et al. 2003). In addition, we overcame the drawback of the scarce shape tuning capabilities by introducing the aesthetics fragment to interact with stylists. In fact, in the ontology we developed a set of aesthetic operators that manipulate aesthetic properties and are mapped into curve operators of the shape grammar.

3 Structure of the ontology

In essence, the aesthetic design activity is the design phase where the character of a product, i.e. the impression that the designer wants to imprint to a product, is expressed. At this stage, sketching, which is a 2D activity and fundamentally a curve-based one, is still the common approach to achieve the previous goal. The resulting sketch is the first materialisation of the designer's mental representation of the product. It is critical to capture the designer's intent at this stage to structure the product data (Cheutet et al. 2005). In the context of car design, the analysis of the designer's know-how revealed that the content of a sketch, based on specific lines and their relative position, effectively contributes to the characterisation of a product. It is therefore desirable to capture the semantics of such leading curves.
The proposed ontology –has been created with Protégé (Protégé), an ontology editor for the OWL language of the W3C consortium (OWL); it addresses two different fields of knowledge (see figure 1). The first captures the aesthetic key elements in car design (which is a more restricted context than the design of other kinds of product). The latter is related to curve manipulation and is more generic, being not limited to designers but addressing also general 2D sketching of contours. These two aspects form two complementary levels in the ontology discussed here,
From the users' point of view, the ontology includes the taxonomy of the ***Aesthetic Key Lines*** (***AKLs***), and the aesthetic properties of such lines. From the geometrical point of view, the ontology contains a 2D curve grammar providing a description of the curve geometry in terms of its curvature extrema and high-level operators to act qualitatively on them, avoiding the manipulation of low-level geometric parameters. It is important to highlight that, even if two fields of knowledge are represented inside the ontology, only the aesthetic fragment will be seen by the end-users, while the second one will be used only internally by the system.
Such an ontology can be associated to a shape modeller so that the geometric modifications aimed at expressing a change of the stylist's intent can be performed consequently. To this purpose, another layer of knowledge will be incorporated and treated: the one translating the 2D curve grammar into the typical low-level parameters of a digital CAD representation, e.g. B-Spline control points and knots. In this way, the preservation or the modification of the aesthetics of a shape may be achieved more intuitively for the user. In other words, designers may rely on a framework conforming to their knowledge, their way of generating, manipulating, interpreting curves, meanwhile producing geometrically sound results.
Figure 2 shows a possible application scenario where the redesign from a C4 to a C4 coupé passes through a modification of some aesthetic key lines. In this case, the process starts from the initial shape on which stylists perform some small modifications to give a more sportive impression but without modifying the identity of the car. To perform this operation, stylists have classically to manipulate the low-level geometry, e.g. the curve control points. On the contrary, here users may act on some aesthetic properties of the curve, which are more directly connected with their design intent, while the design tool will perform the corresponding geometric manipulation.

In the following sections, all the concepts written in bold and italic will be classes of the ontology. The relationships (i.e. object slots inside the ontology) between two concepts will be represented underlined and in italic.

4 Aesthetic fragment

As previously said, even if the final result of the design process is the complete and detailed definition of the surfaces representing the final shape of the product, the character evaluation and modification is performed by concentrating on specific curves of the object, such as profiles, sections and reflection lines (Catalano et al. 2002). Such curves can be both real and virtual: in fact, they can be part of the contour such as profiles and sections, but may also be reflection lines, or more generally lines related to the smoothness of the surface. In order to formalise some typical qualities of a car, a taxonomy of the key curves candidate to elicit some emotions and their properties is proposed here. However, single curves are not usually enough to express a character and it is more common that some properties of specific curves together with special relationships among such curves define the predominant character of a car. The aesthetic relationship between curves can be of two types:

- Geometric relations between adjacent curves: in this case, the aesthetic effect is given by the type of connection between the two curves (e.g. kind of blending, kind of radius),
- Geometric relations between not adjacent curves: in this case, the aesthetic effect is given by the mutual position between the two (e.g. parallelism, angle of incidence, symmetry).

To group simpler the aesthetic key lines, we consider sketches showing the 2D views used in practice. We then subdivide curves according to three main projection views: in fact, the side first and the back/front views are the most important to show the character.

Formally, the most general class in the ontology is the ***AestheticKeyLine*** (fig. 1): it is composed of three subclasses representing the three projection views: ***SideViewAKL***, ***BackViewAKL*** and ***FrontViewAKL***. Each of these classes has two subclasses, one for the profile, and the other for the character line; the former are the curves which belong to the car contour, while the latter groups all the other curves, both real and virtual, contributing to the car aesthetics. For instance, ***SideProfileLine*** and ***SideCharacterLine*** are subclasses of ***SideViewAKL***. Moreover, in the profile category the different portions of the external contour of the sketched car have been named and put in an adjacency sequence.

For each of these classes, a further classification is provided. In particular, the roof line, the windshield line and the wheelbase line in the side view are the most significant. The roof line and the wheelbase line are the first curves sketched by the car designer, just after the wheels. In fact, they both identify the packaging and start to suggest the style. On the other hand, the windshield line with its slope and length contributes to the definition of the aesthetic and aerodynamic quality of the vehicle shape. Correspondingly, in the ontology the classes ***SideRoofLine***, ***SideWheelbaseLine*** and ***SideWindshieldLine*** are represented. In the character line category the waist line and the accent line have particular relevance for the car style; the waist line (represented by the class ***SideWaistLine***) is a curve defining the change of the material between the auto body and the glass of the windows; it is often coupled with the accent line (represented by the class ***SideAccentLine***), a virtual line which expresses the reflection of the light on the surface. The accent line is related to the curvature of the surface in the surroundings of the waist line; it can be also represented by a sharp line with a strong aesthetic effect.

In all the views, some optional peculiar brand lines (represented by the class Side(Front/Back)BrandLine) can elicit the family feeling in order to make a car company recognisable directly from the shape of the vehicle. In some cases, as for Alfa Romeo, the character lines on the hood converging to the logo are definitely brand lines, together with the shape of the triangular front grille. The shape of the light contours

(***Side***(***Front***/***Back***)***LightContour*** class) has also a visible influence on the global character of the car. In Figure 3, the main aesthetic key lines are shown.
Since the AKL is a 2D curve, the class ***Curve2D*** has been introduced and the relation *hasGeometry* has been added from ***AestheticKeyLine*** to ***Curve2D***: the property links the semantics of the curve with the geometric representation, and it is functional (i.e. a AKL must always have only one geometric representation).
Once defined the taxonomy of the aesthetic key lines, the ***Aesthetic Properties*** (***APs***) of such lines, which apply to AKLs, have to be defined. They are naturally related to the geometry, but in a complex way, and reflect the aesthetics of the shape. In this paper, we concentrated on the properties identified as the terms used by stylists for expressing desired shape modification in the FIORES-II project, where their definition and measurement have been finalised and validated working in close collaboration with stylists. In particular, the concepts of acceleration, softness/sharpness, tension, convexity/concavity, flatness, crown have been specified together with their measures. In Section 6, we will only summarise the definitions (details can be found in Giannini et al. (2006)), which will be then translated in terms of grammar operators.
In the aesthetic fragment, the class ***AestheticProperty*** is present and further subdivided; in addition, the relation *hasProperty* has been created from the class ***AestheticKeyLine*** to the class ***AestheticProperty*** (fig. 1).

5 Curve description fragment

To realise high-level operators for an effective curve manipulation, in the ontology we introduced a specific fragment for a curve description based on the extension of the process grammar defined by Leyton (1988) and the specification of a set of manipulation operators.
The Leyton's grammar addresses 2D C^2 curves without self-intersection; here only its basic concepts are briefly recalled in function of its extension with quantitative operators, being directly related to some aesthetic properties. The adopted grammar is based on the curvature, intrinsic curve property, and in particular, on the curvature extrema (represented by the class ***CurvatureExtremum***) and inflection points (fig. 1 & 4). These are called characteristic points (represented by the class ***CharacteristicPoint***) and are indicated as follows:

- M+ for a positive maximum (***PositiveCurvatureMaximum*** class),
- m+ for a positive minimum (***PositiveCurvatureMinimum*** class),
- M- for a negative maximum (***NegativeCurvatureMaximum*** class),
- m- for a negative minimum (***NegativeCurvatureMinimum*** class), and
- 0 for an inflexion point (***InflexionPoint*** class).

A differential symmetry axis, identified by the PISA analysis (Process-Inferring Symmetry Analysis) is associated to each characteristic point. It is defined as the locus of points O, which are the midpoints of the arc AB of a circle moved along the shape and always tangent to the shape at these two points A and B (fig. 4.c). This axis is fundamental to specify the direction of the defined operators acting on the characteristic points. In this ontology fragment, the relation *hasSymmetryAxis*, which is a functional relation, links the ***CurvatureExtremum*** superclass to the ***SymmetryAxis*** class.
A curve is described by a name, the sequence of the characteristic points constituting the grammatical description of a class of curves. For example, an ellipse is described by the name "M+m+M+m+". There follows that all the ellipses belong to the same class, which also contains other types of curves. Codons, defined as subsets of the name composed by three consecutive characteristics points, play a special role. Analogously, the classes ***Name*** and ***Codon*** have been created and connected with the relation *containsCodons*; moreover, the relation *isComposedBy* links ***Name*** and ***Codon*** to ***CharacteristicPoint*** and the relation *hasForName* links ***Curve2D*** to ***Name*** (fig. 1).

The grammar is also composed of six grammatical operators that modify the name of a curve, i.e. they act on a curvature extremum or an inflection point to generate a new codon. With these operators, a user can always transform one class of shapes into another one. The six grammatical operators are:

- CM- (continuation on M-): pushing on a M-, it becomes a M+, with the creation of two inflection points: M- → 0M+0 (fig. 5.a);
- Cm+ (continuation on m+): pushing on a m+, it becomes a m-, with the creation of two inflection points: m+ → 0m-0 (fig. 5.b);
- BM+ (bifurcation on M+): the pushing process on M+ creates a bifurcation creating a lobe: M+ → M+m+M+ (fig. 5.c);
- Bm+ (bifurcation on m+): a protrusion is created into a squashing: m+ → m+M+m+ (fig. 5.d);
- Bm- (bifurcation on m-): the pushing process on m- creates a bifurcation creating a bay: m- → m-M-m- (fig. 5.e);
- BM- (bifurcation on M-): an inlet is created into an internal resistance: M- → M-m-M- (fig. 5.f).

By definition, the modifications allowed by such operators follow the direction of the axis of symmetry, which can be interpreted as the principal direction along which processes most likely evolve or evolved. These six operators do not permit to transform a shape inside a class: in other words, the operators cannot be used to modify a curve from one class to obtain one other curve of the same class. However, the application scenario presented here definitely requires parameters able to manipulate and tune quantitatively a shape with a given name. For this reason, we completed the Leyton's shape grammar by adding some quantitative characteristics and quantitative operators able to distinguish curves of the same class and manipulate them (Cheutet 2006).

Quantitative characteristics will be named *direct*, when they are directly measured on the curvature plot; the others are *indirect*, since they are the result of either certain operations on previous characteristics or off-line computations. The direct quantitative characteristics are (fig. 6.a):

- (Left and right) curvature variation $\Delta k_X(CP)$ with respect to a characteristic point CP, where X designates either the left L or the right R evaluation direction. The left (resp. right) value is determined by the difference between the curvature value of CP and the characteristic point on the left (resp. right) according to the curve parameterisation;
- (Left and right) distance $D_X(CP)$ between a characteristic point CP and the one on the left (resp on the right), computed on the curve;
- (Left and right) range of influence $R_X(CP)$ of a characteristic point CP (fig. 6.b). To determinate the left (resp. right) range, we take the average of the left (resp. right) curvature variation and we search for the point P_X on the left (resp. right) of CP having curvature value equal to $(k(CP) \pm \Delta k_X(CP)/2)$. The curve portion between P_X and CP is the left (resp. right) range of the characteristic point. It can be noted that the union of all the ranges of characteristic points cover the entire curve (fig. 6.c).

The indirect quantitative characteristic is the (left and right) visibility $Vis_X(CP)$ of a characteristic point CP (fig. 7). This characteristic is used to classify the details, considering that a characteristic point is a detail if it has both a small distance and a small curvature variation. It is calculated as the product of the distance $D_X(CP)$ and the curvature variation $\Delta k_X(CP)$ at that CP. A left (resp. right) visibility characteristic can be also applied to a codon and its value in this case is directly the value of the left (resp. right) visibility of the middle codon characteristic point.

In the ontology, the ***QuantitativeProperty*** class is composed of two subclasses: the ***DirectQuantitativeProperty***, further subdivided into ***LeftDistance***, ***RightDistance***, ***LeftCurvatureVariation***, ***RightCurvatureVariation***, ***LeftRangeOfInfluence***, and ***RightRangeOfInfluence***, and the ***IndirectQuantitativeProperty***, composed by ***LeftVisibility*** and ***RightVisibility***. The relation *hasQuantitativeProperty* links ***CharacteristicPoint*** to ***QuantitativeProperty*** (fig. 1).

After introducing the quantitative properties, new operators are needed to act on them. Moreover, in the perspective of defining intuitive operators, we devised a couple of operators not introducing new characteristic points but working directly on curvature value of the CP (Cheutet 2006): in fact, a direct manipulation of the quantitative characteristics has not easily predictable effects on a curve shape. Coupling the grammatical and the quantitative operators enables any deformation of a shape into another, providing designers with extremely flexible manipulation tools.
The quantitative continuation operator is directly inspired from the grammatical one and is analogously indicated as C*X, where X is the curvature extremum which the operator is applied to. A pushing process is applied on a curvature extremum (M+, m+, M- or m-) along the extension of the symmetry axis, but without changing the name of the curve (fig. 8.a). According to the type of curvature extremum the operator is applied, the quantitative properties are modified differently:

- C*m+ (quantitative continuation on m+): m+ is pushed such that it tends to be an inflection point, i.e. its curvature tends to 0 (fig. 8.a);
- C*M- (quantitative continuation on M-): the symmetric operator of C* m+ (fig. 8.d);
- C*M+ (quantitative continuation on M+): M+ is pushed, increasing its curvature value. This operator affects the quantitative characteristics of the characteristic points adjacent to the M+: for example, if an m+ belongs to the neighbourhood, its curvature value tends to 0 (fig. 8.a);
- C*m- (quantitative continuation on m-): the symmetric operator of C*M (fig. 8.c).

The previous operators work in the direction of the axis of symmetry. But to enlarge the set of the possible configurations, the displacement operator of a curvature extremum in any direction DX (where X is the curvature extremum which the operator is applied to) has been added (fig. 8.b).
Analogously, each of the production rules and the quantitative operators is represented as class inside the ontology: ***CurveOperator*** is the superclass of:

- ***GrammaticalOperator***,
  - ***Bifurcation***,
  - ***GrammaticalContinuation***,
- ***QuantitativeOperator***,
  - ***CurvatureExtremumDisplacement***,
  - ***QuantitativeContinuation***.

The functional relation *actsOnCharacteristicPoint* links ***CurveOperator*** to ***CharacteristicPoint***. The relation *actsOnQuantitativeProperty* links ***QuantitativeOperator*** to ***QuantitativeProperty***. The relation *returnsCodons* links ***GrammaticalOperator*** to ***Codon*** and the relation *returnsGramForm* and its inverse *isModifiedBy* link ***GrammaticalOperator*** with ***Name*** (fig. 1).

6 Aesthetic operators

This section describes how some of the high-level aesthetic operators are translated into sequences of operators of the shape grammar. In the ontology, this will be formalised with the relation *isTranslatedInto* which links the class ***AestheticOperator*** to the class ***CurveOperator***. This relation has a set of sub-relations, which are restricted relations from a specific subclass of ***AestheticOperator*** to a specific subclass of ***CurveOperator*** (fig. 1). As an example, the relation *sharpIsTranslatedInto* is defined as sub-relation of *isTranslatedInto*, and links the class ***SharpOperator*** to the classes ***QuantContMaxPos*** and ***QuantContMinNeg***, according to the specific relations discussed in section 6.2.

6.1 Straight operator

In aesthetic design, such an operator makes a line flatter: that means it does not create a straight line but it deforms a curve to tend to a straight line, i.e. with low curvature values and as few as possible inflection points. Therefore, this operator at first has to remove possible undulations along the curve and then it has to act on the curvature value (fig. 9.a).
Translating it in terms of the grammar elements, an undulation corresponds to a codon 0X0, where X can be M+ or m-. Therefore, eliminating the undulation corresponds to reducing the complexity of the name of the curve by using the inverses of grammatical bifurcation and continuation. The undulations selected to disappear are the codons having the smallest visibility (see section 5).
On the resulting curve, the second step of the straight operator tends the curvature value of the characteristic point X towards zero, by applying the inverse of the quantitative continuation operator C*X (where X can be M+ or m- according to the initial shape).
In case of curves with no inflection points, the straight operator simply reduces to this second step.
As an example, in figure 9.a, the upper curve has name “m+M+0m-0M+m+”, the codon “0m-0” is the one with the smallest visibility. The inverse of the process Cm+ is performed, to obtain the green curve, with name “m+M+m+M+m+”. The green curve is still a noisy one, since its name is complex, i.e. it contains more than three characteristic points, thus the inverse of the process BM+ is applied on the curve to obtain the blue one. The result of this first step is a curve with only three curvature extrema “m+M+m+”, where the two m+ are at the boundaries of the curve, as it is for the yellow curve in figure 9.a.

6.2 Sharp/soft operator

The term softness and sharpness represent one the opposite of the other and are used to describe the properties of transitions between curves or surfaces. When the transition is very fast, i.e. with high curvature variation, it is judged sharp; otherwise it can be called soft. The judgment also depends on the size of the parts and on the distance from which the part is observed, but here we are interested in the operators modifying the quality and not in measuring it.
The operator acts on a curve segment connecting two regions of small curvature, and in practice, it acts on an M+ or an m-. Making a blending sharper means increasing the prominence of a corner between the connected curves without generating undulations, i.e. new characteristic points cannot be inserted in the curve.
As a consequence, the sharp operator will be directly translated by the operator C*M+ (or C*m-, depending on the initial stage): the curvature value of the characteristic point will increase and its left and right range of influence will decrease (fig. 9.b).

6.3 Convex/concave operator

When designers make a curve more convex (or concave, in the opposite direction), they move towards the enclosing semi-circle; thus, the ideal convex curve is the semicircle, or an arc of circle, if the continuity constraints at the endpoints are compatible with, otherwise it is the curve that satisfies the given continuity constraints and presenting a curvature with no sign changes and with lowest value variation.
At first this operator deletes the undulations in order to obtain a sequence of type “X+Y+X+” or “X-Y-X-”, where X and Y can be M or m with X□Y. Thus, as in the case of the straight operator, it uses the inverse of continuation grammatical operators on each characteristic point having a small visibility.
The second step produces a more symmetric curve equilibrating the left and right distance (from the lower curve to the middle one in the example in figure 10.a): in particular, it applies the displacement operator on the middle codon curvature extremum. Finally, depending on the sequence obtained at the first step, it applies

the inverse of the quantitative continuation operator C*m+ at the m+ critical point(s) in the first case, and the quantitative continuation operator C*M- at the M- critical point(s) in the case of second sequence type.

6.4 Tension operator

Tension has been defined as the "internal energy" of a curve subject to continuity constraints at its boundaries, provided it is not a straight line and has no inflections. In fact curves including some inflection points are generally referred as wet curves or S-shaped. Tension can be geometrically referred to the evolution of the curvature along the curve, in particular increasing the tension of the curve leads to a larger part of small curvature.
Translating it in the grammatical form, tension can be perceived when the curve has one curvature minimum with a small curvature value in-between two curvature maxima and having large left and right distances.
As for the previous operators, the first step of the tension operator is to suppress the undulations. Then, the curve is possibly modified to obtain a name of type “M+m+M+” or “m-M-m-”. After that, “more tension” is obtained by applying the quantitative continuation operator C*m+ (C*M- depending of the initial stage) to decrease the curvature value of the curvature extremum to tend to 0 (fig. 10.b).
This operator can be also applied with tangency continuity conditions at the endpoints of the curve. In this case, the curvature value of the two other extrema will possibly increase.

6.5 Crown operator

Differently from the previous operators that work on some specific aesthetic properties, crown has a more operational aspect not directly connected to any property. The crown operator is mainly used to lift or raise in a given direction a certain part of the curve without changing the end points. This operation is mainly applied on already convex curves and should not increase the complexity of the name of the curve.
This operator can be decomposed as follows, in the extended grammar:

- the first step is to create a “ghost” characteristic point chosen by the user (it will be a generic point on the curve) and to perform a fictitious curvature extremum displacement according to the given direction;
- afterwards, the grammatical operators are used to adjust the curve in order to avoid an increase of complexity of the curve (i.e. a creation of new extrema during the deformation process).

For this operator, the grammatical operators and quantitative ones can be also used to monitor the process, for instance by defining a-priori a range of validity for the operator.

6.6 Acceleration operator

A curve is said to be accelerated when the variation of the tangent is bigger around one end point when moving towards that point. A wet curve, a straight line or a true radius, i.e. a blending with constant curvature, have no acceleration at all. The acceleration operator is meaningful only if applied to curves which have already the acceleration property.
Thus, accelerating the curve towards one end point means increasing the range of influence of the critical point X immediately preceding/succeeding the critical point Y (M+ or m-) closest, possibly coinciding, to the considered extremum. To obtain this result, different approaches can be adopted. One possibility is to apply simultaneously two quantitative operators: the quantitative continuation operator C*M+ (resp. C*m-) to increase the curvature value of the closest curvature extremum Y, and the curvature extremum displacement DM+ (resp. Dm-) to correct its displacement generated by the continuation operator (fig. 11). As a consequence the distance between Y and X and the range of influence of X in the direction of Y will increase.

7 Conclusions

The increasing demand for accessing and sharing digital shapes (such as for collaborative design, on-line training and documentation) enhances the needs of structuring the shape knowledge at any step of the design workflow, thus making a mapping process among the various stages also possible.
This paper moves into this direction, devising an ontology to formalise the knowledge embedded in car styling. It also provides the basic framework of a design environment for 2D digital sketches in which the traditional modelling systems may be completed by semantic-based and context-aware tools; in this way, stylists and engineers are allowed to create and manipulate shapes more intuitively. Such an environment takes advantage of a specific shape grammar able to convert aesthetic shape manipulations into geometric operations.
The presented work concentrated on the conceptualisation of the knowledge domain. The benefits of the aesthetic operators introduced to simplify the shape modelling has been proved by the result of the FIORES II project both in terms of efficiency and efficacy. The implementation of the operators of the shape grammar cannot impact on the efficacy of the approach, since they are embedded in the aesthetic operators, whereas their efficiency cannot be fully validated yes since they are still under development.
As a natural future activity, the deformation engine developed in Cheutet et al. (2005) will be coupled with this aesthetic environment, creating in this way a semantic modeller. The missing step is the formalisation of the geometric representation into the ontology and its connection with the grammatical fragment.

Acknowledgement

This work is currently carried out within the scope of the AIM@SHAPE Network of Excellence (AIM@SHAPE) supported by the European Commission Contract IST 506766.

References

Alessi: www.alessi.com

AAT Art & Architecture Thesaurus Online.
http://www.getty.edu/research/conducting_research/vocabularies/aat/ (accessed 4 December 2006)

AIM@SHAPE: Advanced and Innovative Models And Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects, *European Network of Excellence*, Key Action: 2.3.1.7 Semantic-based knowledge systems, VI Framework, URL: http://www.aim-at-shape.net. (accessed 4 December 2006).

Brunetti, G., Grimm, S., Feature Ontologies for the Explicit Representation of Shape Semantics, International *Journal of Computer Applications in Technology*, 2005, vol. 23, pp. 192-202.

Cagdas G., A shape grammar model for designing row-houses, *Design Studies*, 1997, vol. 17, pp 35-51.

Cai H., He W., Zhang D., A semantic style driving method for products' appearance design, *Journal of Materials Processing Technology*, 2003, vol. 139, pp. 233–236

Case K., Karim M.S.A., Aesthetic considerations in evolutionary computer aided design, *Proceedings of CAID&CD'2005*, May 29 – June 1, 2005 Delft, The Netherlands, pp. 476-481.

Chase S.C., A model for user interaction in grammar-based design systems, *Automation in Construction*, 2002, vol. 11, pp. 161-172.

Catalano, C. E., Falcidieno, B., Giannini, F., Monti, M., A Survey of Computer-Aided Modeling Tools for Aesthetic Design, *JCISE Transactions of the ASME*, March 2002, Vol. 2.

Cheutet, V., Catalano, C.E., Pernot, J.P., Falcidieno, B., Giannini, F., Léon, J.C., 3D Sketching for Aesthetic Design using Fully Free Form Deformation Features, *Computer & Graphics*, 2005, vol. 29-6.

Cheutet, V., Towards semantic modelling of free from mechanical products, *PhD thesis in cotutelle INP Grenoble & University of Genoa*, 2006.

De Luca L., Véron P., Florenzano M., Semantic-based modeling and representation of patrimony buildings, *Workshop towards Semantic Virtual Environments*, 16-18 March 2005, Villars, Switzerland, pp. 27-36.

FIORES, Formalization and Integration of an Optimized Reverse Engineering Styling Workflow, *Project n. BE96–3579*, http://www.fiores.com.

FIORES-II, Character Preservation and Modelling in Aesthetic and Engineering Design, *Project GRD1-1999-10785*, http://www.fiores.com.

Fujita K., Nakayama T., Akagi S., Aesthetics function and geometry with feature-based modelling and constraint management, *proceedings of ICED'99 Munich*, august 1999.

Giannini, F., Monti, M., Podehl, G., Aesthetic-driven tools for industrial design, *Journal of Engineering Design*, June 2006, Vol. 17-3, pp.193–215.

Gruber, T.R., A Translation Approach to Portable Ontology Specification, *Knowledge Acquisition*, 1993, vol. 5, pp. 199-220.

Hsiao S.W., Chan C.H., A semantic and shape grammar based approach for product design, *Design Studies*, 1997, vol. 18, pp. 275-296.

Hsiao S.W., Wang H.P., Applying the semantic transformation method to product form design, *Design Studies*, 1998, vol. 19, pp. 309-330.

Kopena J.B., Regli, W.C., Functional Modelling in Engineering Design for the Semantic Web, *In IEEE Data Engineering Bullettin*, 2003, Vol. 26-4, pp. 55-62.

Kitamura, Y., Mizoguchi, R., Ontology-Based Systematization of Functional Knowledge, *Journal of Engineering Design*, 2004, vol. 15-4, pp. 327-351.

Leyton, M., A Process-Grammar for Shape, *Artificial Intelligence*, 1988, vol. 34, pp. 213-247.

McCormack J.P., Cagan J., Speaking the Buick language: capturing, understanding, and exploring brand identity with shape grammars, *Design Studies*, 2004, vol. 25, pp. 1-29.

Miura K.T., A general equation of aesthetic curves and its self-affinity, *in Computer-aided design & applications*, 2006, Vol. 3 (1-4), pp. 457-464.

Mizoguchi R., Kozaki K., Sano T., Kitamura Y., Construction and deployment of a plant ontology, *proceedings of 12th International Conference EKAW2000*, 2000, pp.113-128.

Patil L., Dutta D., Srram R., Ontology formalization of product semantics for product lifecycle management, *DETC2005, ASME*, 2005 September 24-28, Long Beach California USA.

Pernot, J.P., Guillet, S., Léon, J.C., Falcidieno, B., Giannini, F., Interactive operators for Free Form Features manipulation, *book of the SIAM Conference on Geometric Design & Computing*, 2003, Seattle.

Piegl L.A., Ten challenges in computer-aided design, *Computer-Aided Design*, 2005, vol. 37-4, pp. 461-470.

Poitou, J.P., Emotion, Aesthetic, Geometry relationship analysis. FIORES II–WP2-T2.3 Summary Report – GRD1-1999-10385, March 2002.

Posada J., Toro C., Wundrak S., Stork A, Ontology Supported Semantic Simplification of Large Data Sets of Industrial Plant CAD Models for Design Review Visualization, *Proceedings of Knowledge-Based Intelligent Information and Engineering Systems*, 2005, Lecture Notes in Computer Science 3683, Berlin, Heidelberg, New York: Springer Verlag, pp. 184-190.

Protégé Ontology Editor. http://protege.stanford.edu/ (accessed 4 December 2006)

Smyth M., Edmonds E., Supporting design through the strategic use of shape grammars, *Knowledge-Based Systems*, 2000, vol. 13, pp. 385-393.

Stiny, G., Introduction to Shape and Shape Grammars, *Environment and Planning*, 1980, vol. B 7, pp. 343-351.

van Der Vegte W., Kitamura Y., Koji Y., Mizoguchi R., Coping with unintended behavior of users and products: ontological modelling of product functionality and use, *In Proceedings of CIE 2004: ASME 2004 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2004.

Wielinga B.J., Schreiber A. T., Wielemaker J., Sandberg J.A.C., From Thesaurus to Ontology, *Proceedings of the first Int. Conf. on Knowledge Capture (K-CAP'01)*, October 22-23, 2001, Victoria, British Columbia, Canada.

The Web Ontology Language (OWL). http://www.w3.org/2004/OWL/ (accessed 4 December 2006)

Yanagisawa, H., Fukuda, S., Global feature-based interactive Reduct Evolutional computation for aesthetic design, *Proceedings of ASME DETC'04*, 2004, Salt Lake City, USA.

Figure 1. Global overview of the structure of the ontology.
Figure 2. (a) C4 coupé (courtesy of Citroen) (b) modification of the character lines: in blue, C4 AKLs, in red C4 coupé AKLs.
Figure 3. Aesthetic key lines of the treatment (courtesy of Citroen).
Figure 4. a) Grammatical description of a smooth curve, b) curvature plots, c) definition of the symmetry axis.
Figure 5. Examples of grammatical operators: a) CM-, b) Cm+, c) BM+, d) Bm+, e) Bm-, f) BM-.
Figure 6. a) Curvature variation and distance between characteristic points, b) Range of a characteristic point, c) Covering of the curve by the ranges of characteristic points.
Figure 7. The small visibility of the two enlightened characteristic points (on left) characterises a small undulation on the curve (on right).
Figure 8. a) Example of quantitative continuation operator, C*m+ on the top of the curve, C*M+ on the bottom b) Example of the displacement operator DM+, c) example of quantitative continuation operator C*m-, d) example of quantitative continuation operator C*M-.
Figure 9. a) Examples of straight lines, with their curvature plot (straighter from the top to the bottom), b) example of sharp operator (sharper from the left to the right).
Figure 10. a) Example of convex operator (more convex from the bottom to the top), b) curves and their curvature plot, with tension increasing top-down.
Figure 11. Curves and their curvature plot, with acceleration increasing top-down.

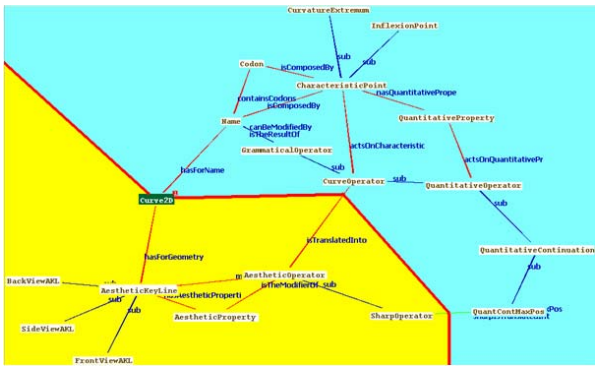

Figure 1. Global overview of the structure of the ontology.

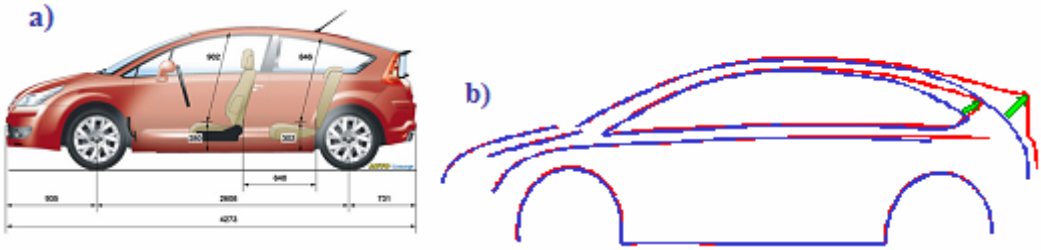

Figure 2. (a) C4 coupé (courtesy of Citroen) (b) modification of the character lines: in blue, C4 AKLs, in red C4 coupé AKLs.

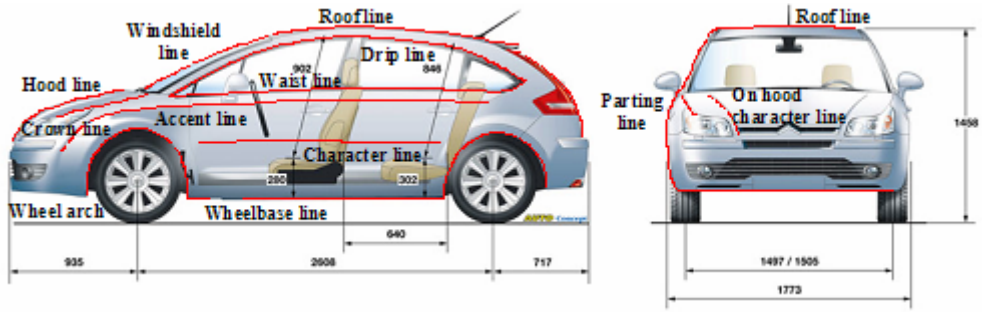

Figure 3. Aesthetic key lines of the treatment (courtesy of Citroen).

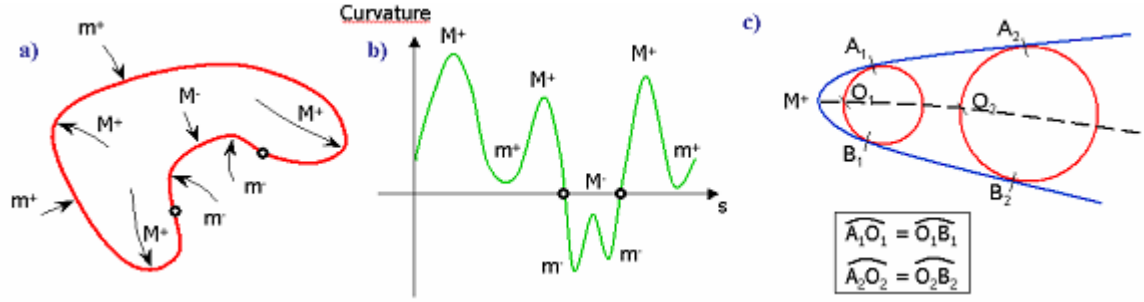

Figure 4. a) Grammatical description of a smooth curve, b) curvature plots, c) definition of the symmetry axis.

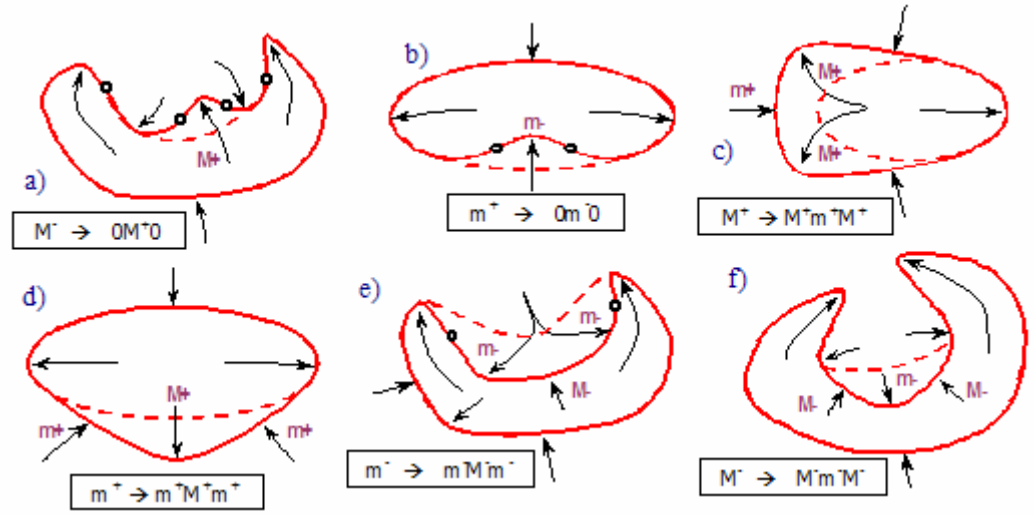

Figure 5. Examples of grammatical operators: a) CM-, b) Cm+, c) BM+, d) Bm+, e) Bm-, f) BM-.

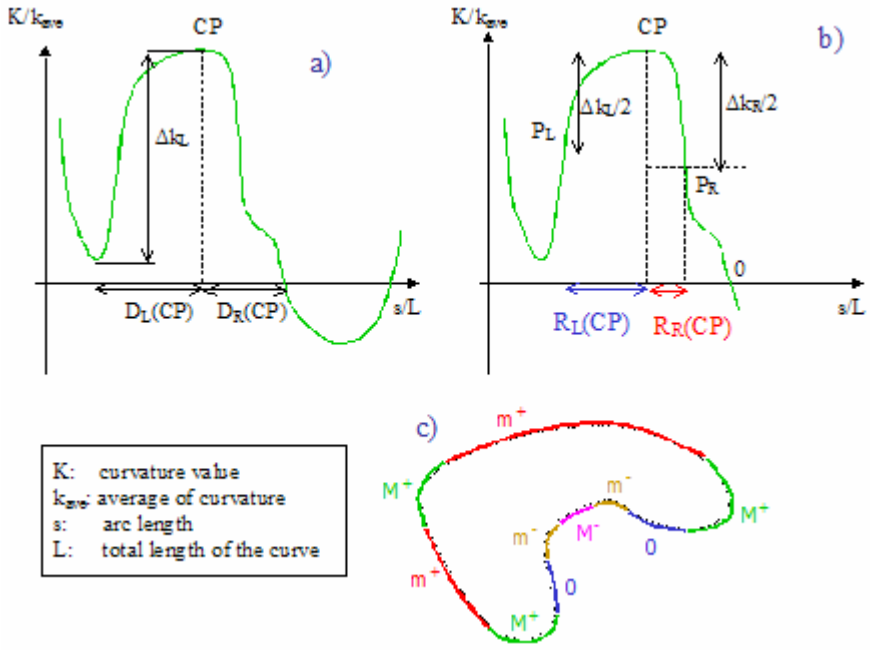

Figure 6. a) Curvature variation and distance between characteristic points, b) Range of a characteristic point, c) Covering of the curve by the ranges of characteristic points.

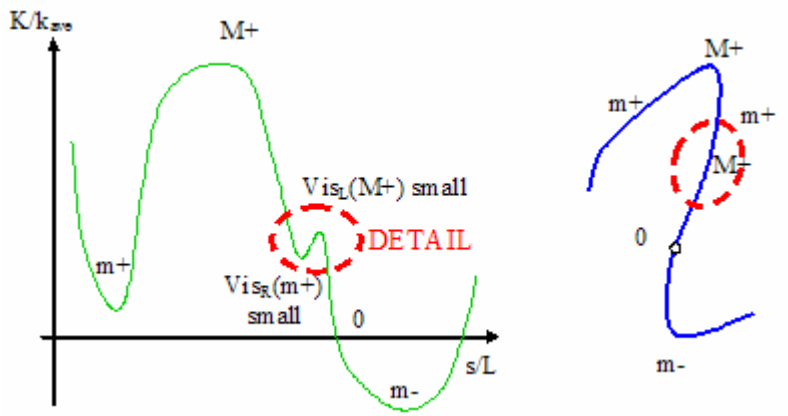

Figure 7. The small visibility of the two enlightened characteristic points (on left) characterises a small undulation on the curve (on right).

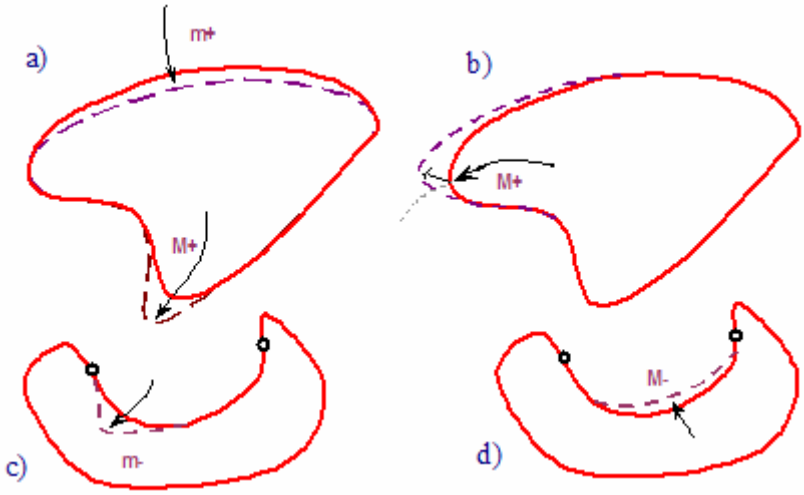

Figure 8. a) Example of quantitative continuation operator, C*m+ on the top of the curve, C*M+ on the bottom b) Example of the displacement operator DM+, c) example of quantitative continuation operator C*m-, d) example of quantitative continuation operator C*M-.

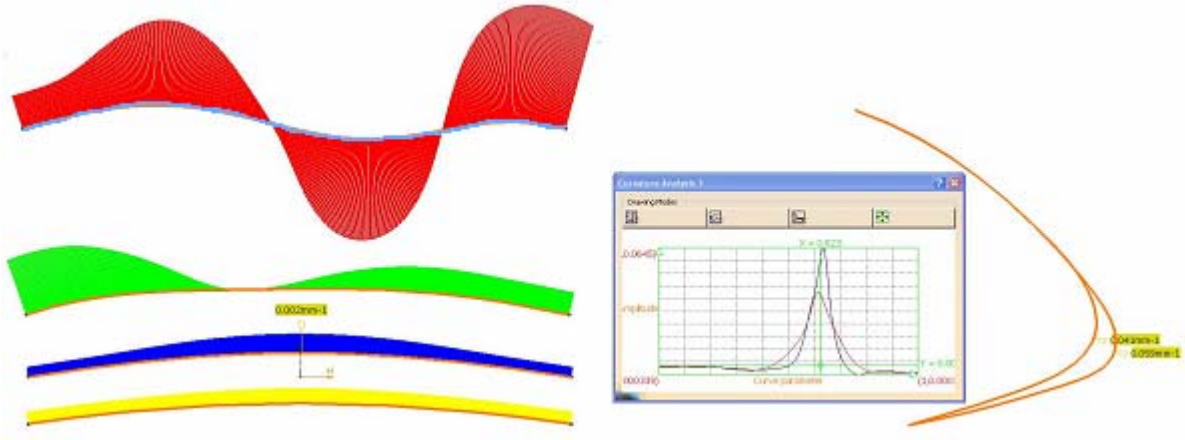

Figure 9. a) Examples of straight lines, with their curvature plot (straighter from the top to the bottom), b) example of sharp operator (sharper from the left to the right).

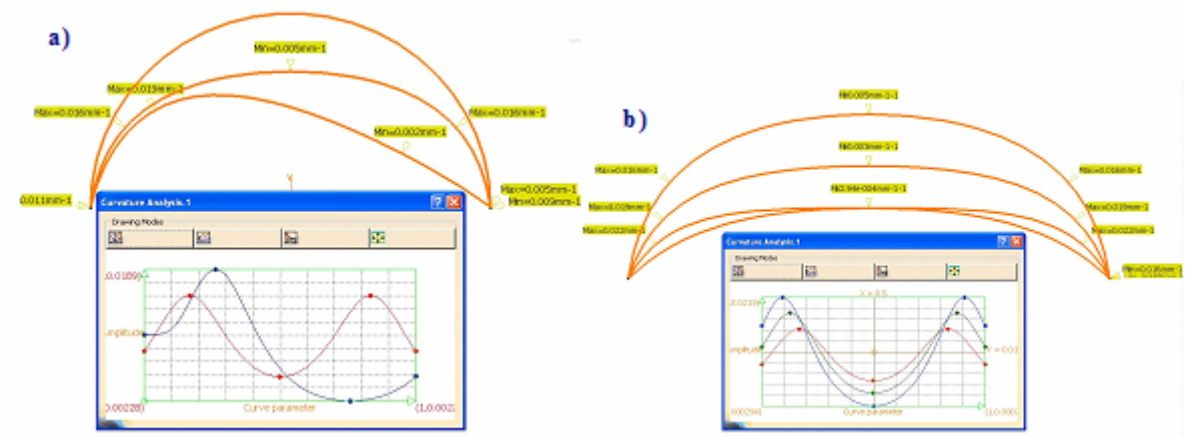

Figure 10. a) Example of convex operator (more convex from the bottom to the top), b) curves and their curvature plot, with tension increasing top-down.

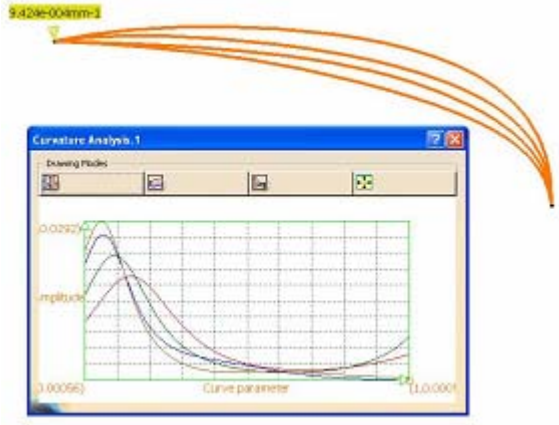

Figure 11. Curves and their curvature plot, with acceleration increasing top-down.