Shape interpretation with design computing

Conference or Workshop Item

Shape Interpretation with Design Computing

Iestyn Jowers
Technische Universität München, Germany

Chris Earl
The Open University, UK

How information is interpreted has significant impact on how it can be used. This is particularly important in design where information from a wide variety of sources is used in a wide variety of contexts and in a wide variety of ways. This paper is concerned with the information that is created, modified and analysed during design processes, specifically with the information that is represented in shapes. It investigates how design computing seeks to support these processes, and the difficulties that arise when it is necessary to consider alternative interpretations of shape. The aim is to establish the problem of shape interpretation as a general challenge for research in design computing, rather than a difficulty that is to be overcome within specific processes. Shape interpretations are common characteristics of several areas of enquiry in design computing. This paper reviews these, brings an integrated perspective and draws conclusions about how this underlying process can be supported.

Introduction

Throughout a given design process, shapes are used in countless ways, and the different uses require different representation schemes to support necessary operations [1]. For example, shapes are used to represent the status of a design concept, as boundary objects that inform communication about a design, as models for analysing the performance of a design, and as instructions for physically realising a design. Often, the shapes in question

Design Computing and Cognition DCC'12. J.S. Gero (ed), pp. xx-yy.

are different views of a single design concept and before they can be used, interpretation into a suitable representation is necessary. Also, during a design task, shapes are interpreted and transformed according to the task's requirements. Information in shapes that is relevant to the task at hand is recognised, and then acted on.

In this paper, two general modes of shape interpretation are identified and explored: interpretation which is *visual*, and informs human performance in design processes, and interpretation which is *analytical* and informs transformations of descriptions used in computational methods. In both these modes, interpretation is concerned with explaining a shape either by applying a meaning or by identifying its structure or parts. In design research, the two modes are typically not explored in parallel, and instead investigations take place within localised contexts, such as conceptual design [2], or CAD/CAM [3]. However, they share strong commonalities and this paper aims to establish interpretation as a general problem for design computing, one that is common across design processes, rather than a local problem that is directly linked to specific contexts.

To this end, the paper presents a review of design research, set within a framework of visual and analytic modes of interpretation, with an emphasis on how humans and machines interpret shapes and apply these interpretations in subsequent operations. It aims to provide a general description of the role that shape interpretation plays, and highlight key similarities between different processes of design. These include the need to manage ambiguity and support the unexpected in design representations, and the importance of context and intended use in driving shape interpretation.

Part 1: Visual Interpretation

As a visually creative activity, design is dependent on processes of perception – the shapes that surround designers inform and inspire them as they undertake design tasks. It is generally suggested that shapes are recognised and interpreted via decomposition into structural parts or features [4]. Understanding of a shape necessitates recognition of its parts and without this a given shape is an abstract entity void of meaning [5]. However, any given shape can give rise to countless decompositions into parts, and consequently countless interpretations. Also, these interpretations are susceptible to change from moment to moment; Wittgenstein [6] describes such interpretations as hypotheses regarding the structure of a shape, which may turn out to be false and are susceptible to change based on newly acquired evidence, or on the viewer's whim. Experienced designers learn to interact with shapes in this way, to visually explore alternative interpretations and structures [7], and this interaction has been linked to innovative design [8].

Interpretation in Conceptual Design

The role of shape interpretation in conceptual design is well documented: shapes are used to form and inform representations of emerging design concepts. Typically such shapes are externalised using sketches, models, gestures, prototypes, digital tools, or verbalisations [9]. However, they are predominantly represented as sketches which are used to support shape exploration by representing particular aspects of a design concept. In this role, sketches are more than just static representations of imagined concepts; they externalise designers' cognitive activity and are used as devices to support exploration of an emerging design [10].

The shapes represented in sketches are inherently ambiguous, and this leads to a rich interaction between the designer and the shapes, what Schön and Wiggins [11] refer to as a reflective conversation between the designer and the media with which they are working. When shapes are viewed as abstract, ambiguity suggests alternative parts and structures that give rise to potentially countless interpretations [7]. When they are viewed as representing a concept, ambiguity enables designers to read off more than they put in [12]. Ambiguity makes it possible for the viewer to hypothesise about the meaning of a sketch, to interpret it based on context or according to their own knowledge and experiences. It allows designers to bring new insights into exploration process and supports the evolution of a design concept.

The kinds of interpretation used across conceptual design are varied. A given shape could give rise to figural interpretations; these are concerned with gestalts – coherent wholes that are defined by viewers' interpretations of the geometric elements that compose design representations. For example, the shape in Figure 1 could give rise to figural interpretations as architectural plans or arrangements of tiles. Alternatively, a shape could give rise to other forms of visual interpretations. For example, the shape could be interpreted as a graph representing a schematic abstraction of an object, or as a collection of constructive elements, such as lines, triangles or squares. Alternatively, non-visual interpretations could arise. For example, the shape could be interpreted according to suggested functional properties, or could act as a metaphor for some alternative meaning or philosophy, such as motherhood or unity.

Studies of conceptual design identify the roles that these different kinds of interpretations play in design exploration. For example, gestalts result from interpretations that assign physical meaning to the geometric elements that compile shape representations, and are not fixed. The same set of geometric elements can be reconstructed as many different coherent wholes, and a designer often shifts gestalt during design exploration processes [13]. Gestalts enable designers to reason about design problems. Similarly, metaphorical interpretation is concerned with analogy – with creating a link from one concept to a (possibly indirectly) related second concept [14]. In conceptual design, metaphors enable designers to apply knowledge from a known situation to an unknown situation; they aid in the structuring of design problems, can contribute to unconventional thinking and stimulate innovation in design activities [15]. Analogies result from interpretations that assign comparative meaning to the geometric elements that compile shape representations.

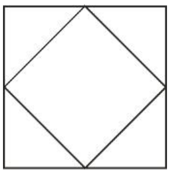


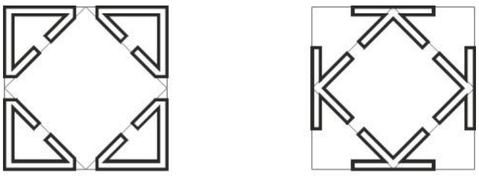
Fig1. An ambiguous shape

Interpretation in Computer-Aided Conceptual Design

The different kinds of shape interpretation used in conceptual design require methods of computational support that are not available in commercial computer-aided design (CAD) systems. This is evidenced in studies reported by Goel [10] and Stones and Cassidy [16], where designers undertook conceptual design tasks using either sketching or commercial computational tools. Both studies found that participants readily use shape interpretation in their design exploration if sketching, but not when using computational tools. Stones and Cassidy observe interpretation did take place cognitively when computational tools were used, but there was no evidence of these interpretations in the creation of new solutions. They suggest the reason for this is that, when participants were using computational tools, they were looking for accuracy in their design concepts and until a form closely resembled their mental picture they were unable to progress to alternative interpretations. Lawson and Loke [17] propose a more pragmatic reason and suggest that development of computational design tools has placed too much emphasis on graphical representation techniques. As such, the resulting tools are unable to support processes essential to creative design, including the process of shape interpretation as a means for supporting shape exploration.

Schön [13], discusses the potential for providing computational support for conceptual design, and distinguishes between methods that recognise designers' interpretations and those that support interpretations. This distinction is concerned with the difference between the semantics of shapes and the syntax of shapes. The semantics of a shape reflect the meaning that is associated with it, such as what it represents figurally, functionally, metaphorically, etc. As discussed, these are an important aspect of creative design, and build on designers' knowledge and past experiences – sources of information not necessarily evident in the shapes that are used to support conceptual design, or apparent in the situation in which the process takes place. As such, the cognitive processes involved in this level of (semantic) interpretation are difficult, if not impossible, to formalise using computational methods [18]. Consequently, methods that seek to recognise designers' interpretations, such as Setchi and Bouchard [19], are necessarily restricted with respect to context; they provide only limited allowance for the unexpected and unknown, and as such their capacity for supporting innovative design is questionable.

Implicit in any semantic interpretation of a shape is a syntactic interpretation, i.e. a constructive interpretation of the geometric elements used to structure it [4]. For example, any figural interpretation of the shape in Figure 1 necessitates a supporting syntactic interpretation, as illustrated in Figure 2, where the shape is interpreted as an architectural plan in two different ways. The spaces that these two interpretations define are very different – in Figure 2a, the shape is interpreted as four closed triangular wings overlooking an open quadrangle, while in Figure 2b it is interpreted as a closed square hall with four open vestibules. In both of these examples, the semantic interpretations are implicitly dependent on different, and incompatible, syntactic interpretations of the same underlying shape.



a) b)

Fig2. Interpretations of a shape, from Stiny (private communication)

During exploration processes, designers are continually making syntactic interpretations of the shapes with which they are working, in order to support their semantic interpretations. Schön [13] suggests that it is here that design computing can best support conceptual design by allowing shapes to be manipulated according to the parts recognised at any particular moment. However, supporting syntactic interpretations can be problematic because (1) parsing against syntactic structures is difficult and (2) the directions taken by designers in following different syntactic interpretations are wide ranging and surprising. Indeed, it is generally acknowledged that commercial design systems offer poor support for such syntactic shape interpretation, because the data structures on which they are built assume that a given shape has a unique interpretation [20]. This means designers have to adapt their design practice so that they are consistent with the particular systems that they use (and the underlying data structures), and this is often evident in a lack of innovation in shape exploration [21]. Research that seeks to address this problem has considered how shapes can be represented and queried so that the parts recognised by the viewer are apparent for manipulation, e.g. [7], [20]. Interaction methods that allow designers to intuitively specify their interpretation of a shape at any given moment, according to recognised parts, have also been explored e.g. via sketch-based input [22] or eye-tracking [23].

Interpretation in Collaborative Design

The processes used during conceptual design, to explore and develop design concepts, are not usually conducted by solo designers working in isolation. Instead, it is common for designers to work in teams that develop concepts in collaborative, social processes [24]. The shapes used to support these processes are varied, and include digital models, prints, physical models, flow charts, gestures etc., and these are used in various roles. For example, Ferguson [25] discusses three roles for sketched shapes as media for collaborative design: thinking sketches, talking sketches, and prescriptive sketches. Thinking sketches refer to sketches used in design exploration, and are interpreted in different ways to form and inform design concepts, as discussed above. Talking sketches support design communication in collaborative design. They act as conscription devices, that organise and store knowledge created through group interaction, and as boundary objects that support communication between participants of different disciplines [26]. In this way, shapes foster collaborative idea generation by providing a collective memory and by allowing team members to reflect on and interpret the ideas of other members [27]. Prescriptive sketches are used to record the outcomes of conceptual design, and are used to inform the representations that support downstream processes, such as analysis and fabrication.

The process of collaborative shape interpretation is not straightforward, and introduces additional issues over individual sketching processes. How shapes sketched by one team member are interpreted by others depends not only on form and context, but also on physical actions and social interaction. The process of constructing and transforming shapes conveys important information that is not necessarily apparent in the shapes themselves [28]. Also, when shapes are used to communicate design thinking, the apparent visual ambiguity gives rise to misinterpretations. Complications arise because participants in design teams do not necessarily see shapes in the same way. For example, Maier et al. [29] discuss the difficulties that arise when design engineers and simulation engineers communicate during design processes. Design engineers interpret shapes in terms of apparent geometrical structures, while simulation engineers interpret them in terms of functions, and this leads to difficulty in communication. Specifically, Henderson [26] notes that embedded within shapes are "codes" which are read by different viewers at different levels. These codes act as visual syntax or jargon and are defined within social structures, such as a design disciplines. Obvious examples include the standardised symbols used to annotate mechanical and electrical technical drawings. But, codes can also be more subtly embedded, for example in conventions that define how shapes are constructed and presented. This is illustrated in Figure 3, where the inclusion of a line closing a concavity indicates that the shape represents a cylinder, rather than a flat surface. Codes represent visual languages that are obvious to practitioners of a relevant discipline, but may not be obvious to less-experienced practitioners or outsiders. Therefore a shape which means one thing to one member of a design team may be read differently by other members.

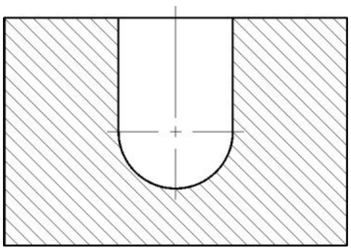


Fig3. Technical drawing of a cylinder, from Henderson [26] (p. 56)

Shah [30] suggests that misinterpretations that arise during design communication can be beneficial to a creative design process, since they can lead to exploration of unexpected ideas. This mirrors the use of interpretation in conceptual design but, for the sake of design communication, misinterpretation is not always beneficial. Indeed, Stacey and Eckert [31] emphasise that design communication should lead to a shared interpretation of a design. Ambiguity that results from incompleteness and provisional decisions is a necessary part of design, and can support a collaborative process. But, ambiguity that is cause by vagueness in shape representation can lead to confusion and should be avoided, since this can result in violation of decisions previously made and the contradiction of requirements or constraints. Similarly, van der Lugt [27] notes that lack of clarity about which parts of a shape are available for interpretation can lead to a lack of creativity in design collaboration. The possibility of disrupting the intentions of team members causes hesitation with respect to interpreting and modifying the shapes created by others. To avoid this, members of design teams seek permission to engage in reinterpretation of colleagues' sketches. Participants in a collaborative design process should clarify where negotiation in a design concept is possible, which elements of a shape are provisional, and which are constrained.

Interpretation in Computer-Supported Collaborative Design

When designers work face-to-face, communication occurs using a patchwork of shape representations and human interactions, including verbal communication and body language. For example, gesture is an important communication device in collaborative drawing activities as a method for sharing interpretation of shapes [32]. Gestures can act as a collective memory and can support collective interpretation, in a manner similar to sketched shapes. But they are rarely used in isolation and instead are used to refer to objects, such as shapes or other members of the design team, and are generally accompanied with verbal explanation. This combination of interactions means that any misinterpretations that occur in face-to-face collaboration can be quickly recognised and corrected. Distributed design teams do not have this richness of representation to work with, and this can have a detrimental effect since the discourse that suggests how shapes should be interpreted is missing [28].

Current computational support for collaborative design does not adequately support these human factors of design communication [33]. 3D virtual worlds seek to address this issue by providing team members with avatars that provide a sense of shared presence [34]. But the interactions that these allow are limited and do not support the richness of face-to-face interaction. Indeed, the communication of distributed design teams is limited to the interactions that their tools allow. Because of this, use of communication tools to support design collaboration, necessarily affects design behaviour. For example, Maher et al. [35] compare design behaviour exhibited in face-to-face collaboration with collaboration in a remote sketching system and collaboration in a 3D virtual world. They discovered that collaboration using communication tools results in less time spent on the design process, more time spent on discussing software features, and a decrease in analysis-synthesis activities. They also found that the shape representations that the tools support modify the designers’ interactions with them.

Part 2: Analytical Interpretation

Problems of shape interpretation are also manifest in the computational methods and tools used to support design processes, such as computer-aided design/manufacture (CAD/CAM) systems. Such tools are developed for specific domains, such as mechanical engineering or architecture, and are used to construct, manipulate and interrogate digital models that represent design concepts. Here, interpretation is distinct from visual perception, and instead is concerned with analysing and transforming the descriptions of shapes so that the structures and parts necessary to carry out specific operations are defined. Problems arise due to the need to transfer data into, and between tools. This is because the different domains and different methods have different requirements with respect to the data used to represent shapes [3]. Integration of tools is highly desirable, since without it data needs to be transferred manually. This can be expensive, both temporally and financially, it is potentially disruptive to the design process, and increases the potential to introduce errors. Also, integration of representations is desirable so that design models can reflect the multiple perspectives, and multiple levels of detail, that are necessary to support multi-disciplinary design.

Interpretation for Design Analysis

Throughout design processes, various methods and tools are used to analyse concepts against domain specific requirements. Central to the effective use of these methods is the problem of reducing the complexity of a design model so that desired properties are readily available for analysis. For example, in mechanical engineering, finite element analysis methods are commonly used to assess the structural properties of a design, such as strength [36]. This is achieved by interpreting the shapes in a model according to a simplified mesh of polygons, a process that is guided by the attributes of the original shape in combination with the specified goal of the analysis. In architecture, ‘walk-through’ and other simulations are used to assess spatial properties, such as ‘flow’ [37]. In this case, simplification of a model is achieved by defining key aspects of the simulation, e.g.

points of interest in a crowd movement simulation or shadows for a light simulation. Similarly, visual aesthetic qualities are analysed, for example to ensure conformity to a brand or style [38]. Such analysis is achieved by identifying characteristic shapes, their allowable variation and the allowable spatial relation between them.

These three examples illustrate a spectrum of analysis problems, differentiated according to how quantifiable they are. Structural analysis exemplifies problems that are fully quantifiable and as such are commensurable with numerical methods of analysis, and implemented in computational systems. Aesthetic analysis exemplifies the opposite end of the spectrum, where problems are very difficult to quantify computationally, and require human interpretation of the results. Simulations lie between these extremes and can be used in distinct ways; firstly, to virtually test designs in use. This depends on human interpretation, and simulations should allow both realistic views by users as well as interpretations by them. The success of analysis can depend on the interfaces, and modes of interaction supported, e.g. the inclusion of user action or tactile feedback [39].

The second, more quantifiable, use involves determining an optimum or 'best' solution within given constraints and resources. In such problems, interpretations are expressed as shape parameters, and the values of these parameters are searched via simulation (or other methods of analysis), within the constraints, and to a given degree of accuracy. The possible designs generated are part of an 'object world' [24] instantiated for a particular project at a specific level of accuracy. These object worlds are context specific interpretations of potential designs which designers manipulate and optimise. Optimisation can be employed in a wider context to search across design schema or configurations [40], as well as instances within a configuration.

Shape interpretations for design analysis do not end with properties and behaviour of the design itself. They are also key properties in the 'design for x' scenarios of manufacture, assembly and fabrication. These may be absolute in that designs may not be physically realisable, or relative in that the necessary resources are not available at the time. For example, if and how design shapes can be constructed or manufactured within cost and resource constraints is a critical analysis, often required at quite an early stage in the design process [41]. But analysis will yield more than a 'go/no go' result; its purpose is to provide routes to design improvement through understanding of possible changes to design components and assemblies. The initial analysis comes through a particular interpretation with incremental changes involving adjustment to this interpretation. It may yield performance outside acceptable margins leading to radical design changes with corresponding new association and analysis.

In the process of analysing designs, the design intentions, expressed through functions and requirements, play an important role. As designs evolve, analysis, assessment and evaluation determine alignment with intentions and requirements, which themselves evolve alongside design development. But functions and associated descriptions are wide open to interpretation themselves. Alink et al. [42] demonstrate the importance of interpretations in functional descriptions of mechanical devices, which are predominantly about shapes of components – their surfaces, interfaces and interstices. Functional descriptions correspond to shape interpretations. The wide variation in the functional descriptions observed in this study shows a broad spectrum of shape interpretations which are possible during design processes. This exemplifies the different perspectives that various people, engineers, technical sales and marketing, for example, will hold. All these interpretations play into a design process and product evaluation. The diversity of descriptions integrated in design again points to the critical role of interpretation.

Interpretation as Feature Recognition

The ways that digital shape data is presented throughout design processes varies. For example shape data may be presented as point clouds from laser scans of prototypes or as CAD surface descriptions from CAD processes. These different object descriptions pose their own issues for shape interpretation. In point cloud scanning there is no inherent surface structure in the acquired model [43]. On the other hand CAD surface descriptions are constructed from a series of shape elements and surface approximations. The scanned data may be grouped together in surface patches but a key issue, whether in point cloud or CAD surfaces is the relation between these geometric elements and the meaningful design and manufacturing features of a product or its components. In both cases this step is an interpretation from data to features.

Features are generic shapes used in computational tools, for supporting multiple shape interpretations. They are meaningful in specific application domains such as design, analysis or manufacturing, and they apply semantics to the shapes in design models, that reflect how those shapes will be understood in a particular process [44]. A given design model can be interpreted according to features in different ways, depending on the semantics that need to be represented. The resulting feature-based models are defined either according to a bottom-up or a top-down approach, using either design-by-features methods or feature recognition, respectively.

In design-by-features methods, features are used directly to construct design models. These features can be defined as shapes with specific significance, such as design components, but are more generally generic shapes, such as cylinders or rectangles [45]. Design-by-features methods allow designers to avoid tedious low-level shape definitions, and support spatial reasoning at a higher level of abstraction, whilst also conveying design intent. However, the set of features on which a particular method is based can never be comprehensive and can never support every conceivable situation. This is likely to feel restrictive to some designers, and has the potential to stifle creativity. Also, as a design is developed and modified, maintenance of features and the semantics linked to these features is a challenge [46].

Feature recognition is the problem of interpreting a given shape according to a defined set of features, i.e. the problem of recognising specific geometric shapes embedded in the representation of a design model [47]. This problem is complicated due to the possibility of multiple solutions, and due to the possibility of partial features (recognised by "hints") which result from the interaction of features. It is a generally unsolved problem, and although various approaches have been defined and successfully applied, they are limited in application. Also, the shapes that can be considered are limited and the recognition of freeform features remains a challenge [48].

Feature-based models are domain specific. For example, the features used by designers to construct a shape are inherently different from the features used to define a process for fabricating the shape, as illustrated in Figure 4. Because of this, features support different views of a product model, and feature-recognition suggests the potential for integrated design models by making available the information that is relevant for different design processes [49]. However, the domain specificity of features raises the question of how different feature-based interpretations of a model relate to each other – this is not always obvious and is of great concern when feature-based models are modified throughout a multi-disciplinary design process.

Feature mapping methods consider the problem of converting a model defined by one set of features into a model defined by a second set [47]. In theory it is a different problem to feature recognition since methods can take advantage of the features that already exist in the representation. However, there is little evidence to suggest that the information that is represented in a feature model for one domain is useful in another domain. Instead methods generally build on the underlying geometry, and integrated design models are defined by considering the mappings between individual features [50]. In this way, it is possible to manage multiple interpretations of a design model by propagating changes across feature models.

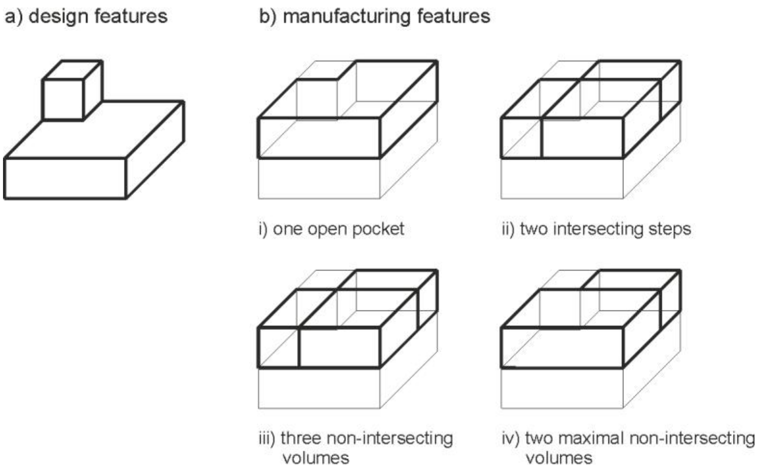


Fig4. Feature interpretations, adapted from Corney et al. [3]

Interpretation for Fabrication

Shape interpretation underlies how designs are evaluated, but designs are not just conceptions and models; their physical construction, through fabrication, demands yet another layer of shape interpretation. Interpreting a model as a specific fabrication processes that can physically realise the design can also be problematic. It is a computationally difficult process, as evidenced by continuing research into process planning, including the CAD/CAM interface [3]. This research aims to make the manufacture of products cheaper in terms of cost and time, by reducing the amount of human input necessary for process planning. This is particularly important as design moves towards a paradigm of mass-customisation and flexible manufacturing systems. Consumers are demanding more individually designed products, and the resultant costs of manufacturing are rising. Flexible manufacturing systems are explored that can provide the required variation at reasonable cost [51]. In particular, autonomous design-to-fabrication systems have the potential to meet the demand for rapid production of high quality products at low cost, by avoiding time-consuming manual re-planning [52].

In CAD/CAM, process planning is supported by interpreting a design as a feature-model, with features defined according to specific manufacturing processes such as milling or casting. Such features support the generation of process plans in computer-aided process planning (CAPP) systems. However, a given design can be interpreted according to manufacturing features in many different ways, as illustrated in Figure 4b, and different interpretations correspond with different manufacturing processes. This leads to the unsolved challenge of determining the best feature interpretation for a given part and a given manufacturing process. The solution is not straightforward, and depends not only on the geometry of the given shape, but also on manufacturing information, such as tool type. Problems such as these mean fully autonomous process planning is still not possible and process planning tools still require an extensive amount of manual input regarding machine types, setups, fixtures, operations, cutting tools, cutting parameters etc. [53].

An advantage to having human input into process planning is that domain experts can control the details of the manufacturing process. Indeed, Corney et al. [3] suggest that automatic feature detection may not be necessary or required, since there is benefit for humans to make some decisions themselves. A less time-consuming approach would be to encourage designers to consider manufacturing processes as they compose a design shape. Features can be used to support 'design for manufacture' philosophies. Manufacturing features can be recognised as the designer creates a design, in order to identify potential manufacturing difficulties and evaluate alternative plans [41]. Alternatively, design-by-feature methods can be used to force designers to construct designs according to manufacturable features that can be easily recognised for process planning [54]. However such an 'object worlds' approach is characteristically deterministic, and is limited in application.

Rapid prototyping techniques, such as fused deposition modelling or selective laser sintering, provide a cost effective alternative to the more traditional methods of design fabrication [55]. They avoid the need to interpret designs according to features and support flexible manufacturing of complex forms, not realisable with traditional methods. This is because they use additive processes, where fabrication of 3D shapes is simplified according to a 2D layering process. There is limited need for CAM or CAPP processes or human intervention since the pre-processing of a shape simply involves tessellating a CAD model so that is can be efficiently interpreted according to horizontal slices. The only variations in fabrication relate to orientation of the design, which influences both build time and surface finish.

The rapidity and low cost with which physical representations of shapes can be produced using RP technologies provide a substantial reduction in product development time. Physical models are included in the shape exploration process; incomplete and provisional models are fabricated, assessed and visually interpreted, much in the same way that sketches are used [56]. They are used to visualise and physically explore concepts, to verify and optimise design parameters, to support iteration of design ideas, and to communicate those ideas with others.

Discussion

Shapes have visual properties that lend an ambiguity and richness to creative processes such as design. These properties give rise to interpretations that inform the agents involved in such processes, by applying meaning (semantics) to the shapes and/or by identifying their parts and structures (syntax). The semantics of a shape are intrinsically linked to the context in which the shape is situated, along with its intended use. For example, figural interpretations of a shape depend on the viewers' understanding of the form it represents (as illustrated in Figure 2), while the features that are important in the shape are dependent on the processes that are to be applied to it (as illustrated in Figure 4). In general, the problem of identifying the context of a shape remains a formidable challenge, akin to the (as yet unmet and possibly unattainable [57]) requirements of strong AI. As such, it is likely that human intervention will always be necessary to guide computational methods with respect to the context and intended use of a shape, and methods of human-computer interface that efficiently and intuitively afford such guidance should be explored. This human intervention is not necessarily undesirable, since it means that there is room for human expertise to inform computational design processes, respond to the unexpected, and resolve potential conflict [3], [31].

The syntax of a shape describes the structure of its representation and while visually it is linked to the semantics (and context) of the shape, in design computing it is separated. It is here that interpretation is a tractable problem for design computing [13], and it is also here that a clear distinction can be drawn between the two modes of interpretation, visual and analytic, that have been highlighted in this paper. Visual interpretation is based directly on perception, and there is no distinction between the visual shape and its representation, i.e. the shape *is* the shape. This means that, when designers use physical media (such as sketches, models, etc.) they are able to take advantage of the visual ambiguity and richness of shape, interpret it according to unexpected forms, varying contexts or intent, and in response, directly modify the shape. However, when computational methods are used to support design processes, the visual shape is a rendition of formal data structures. These structures have been developed based on the underlying problem of how to construct, manage and efficiently render digital models that reflect the forms that are apparent in the natural world. To this end, they succeed in representing highly complex forms with increasing accuracy and speed. But, the visual properties of shapes are not accounted for, since the data structures fix on one interpretation, leaving the ambiguity and richness of the visual shape absent. The shape is *not* the shape, but is a visualisation of a specific data structure. Different data can give rise to the same (visual) shape, and analytical interpretation is concerned with identifying transformations between these. Visual properties are typically not apparent in the data, and this can result in designers having to modify their practice to suit the computational tools that they are using [21]. For example, the geometry that can be used to define shapes might be restricted [54], and/or the transformations that can be applied to those shapes constrained [2]. Designers have to construct digital models in an 'object world' approach to meet a specific and limited purpose [24], and these models cannot be freely interpreted according to unexpected forms, varying contexts or intent.

So, at a fundamental level, the problem lies with the shape representations that have become standard within computational tools, e.g. boundary-representations (B-rep), which build on point-set topologies. In particular, there is a disconnect between the visual shape and the underlying representation. The point-set approach defines shapes according to symbolic structures, ordered according to the relationship of inclusion, and these do not reflect the perceptual characteristics of shapes. They do not afford the multiple interpretations that are needed to support design processes, and neither do they support the examination and re-examination of shapes to identify alternative parts and structures. Instead, alternative shape representations require investigation; representations that will afford development of tools that suit design practice, rather than designers having to modify their practice to suit their tools. For example, Salustri [58] suggests that the logic of mereotopology is a suitable alternative that formally describes "real" entities. In this approach emphasis is placed on the continuity of shapes, and the relationship of part-hood. Shapes are represented as occupying regions of space, and other concepts such as points, boundaries etc., result from interactions between these. There is a long philosophical and mathematical background in such ideas as a foundation for geometry including Whitehead [59], Clarke [60] and Gerla [61]. For design computing, the shape grammar formalism of Stiny [20] is based on a similar premise in which shapes are primarily structured according to parts, identified through the querying mechanism of shape rules. This formalism supports the reinterpretation of shapes, according to parts that are identified and manipulated via the application of such rules. Part-based topologies such as these support interpretation of shapes because they allow the structure of a shape to be defined according to whatever parts are relevant in the current (user-defined) context. These contexts can serve to 'fix' parts in the structure, forming the basis of a semantics of shapes situated in the current design context.

However, as suggested by Sloman [1], it is unlikely that any single representation scheme will sufficiently capture all the ambiguity and richness of shape, and instead it is likely that a variety of types of representation are necessary to support visual interpretation. A similar conclusion is suggested by Hanna [62] who reports that allowing high-dimensional representation of shapes, according to a variety of schemes, enables interpretation of that representation by an artificial agent. In other words, given enough representational data about a shape, relevant characteristics of the shape, such as neighbourhood type, can emerge. Hanna's example relates to the classification of buildings, but a similar approach applied to other problems of shape interpretation may be possible.

The problem of interpretation itself is far wider than that considered here. All information that inputs to, is created in, and is manipulated by design processes is interpreted to accommodate specific uses. Stouffs and Krishnamurti [63] suggest that this general problem of information interpretation shares characteristics with the problem of shape interpretation. Accordingly, investigations into how shape interpretation can be supported can potentially inform the problem of how other forms of design information can be computationally represented and interpreted.

Acknowledgments

Iestyn Jowers' research is supported by CoTeSys, the Cognition for Technical Systems Cluster of Excellence (www.cotesys.org), funded by the Deutsche Forschungsgemeinschaft (DFG).

References

1. Sloman A (1984) Why we need many knowledge representation formalisms. In M Bramer (ed) Research and Development in Expert Systems, (pp. 163-183). Cambridge University Press
2. Prats M, Lim S, Jowers I, Garner SW, Chase S (2009) Transforming shape in design: observations from studies of sketching. Design Studies 30: 503-520
3. Corney J, Hayes C, Sundararajan V, Wright P (2005) The CAD/CAM interface: a 25-year retrospective. Journal of Computing and Information Science in Engineering 5: 188-197

4. Bruce V, Green PR, Georgeson MA (2003) Visual Perception: Physiology, Psychology and Ecology (4th Edition). Psychology Press
5. Krstic D (2004) Computing with analysed shapes. In JS Gero (ed) Design Computing and Cognition '06, (pp. 397-416). Springer
6. Wittgenstein L (1991) Remarks on the Philosophy of Psychology. GEM Anscombe, GH von Wright, H Nyman (eds). Wiley-Blackwell
7. Liu Y-T (1995) Some phenomena of seeing shapes in design. Design Studies 16: 367-385
8. Suwa M (2003) Constructive perception: coordinating perception and conception toward acts of problem finding in a creative experience. Japanese Psychological Research 45: 221–234
9. Wiegers T, Langeveld L, Vergeest J (2011) Shape language: How people describe shapes and shape operations. Design Studies 32: 333-347
10. Goel V (1995) Sketches of thought. MIT Press
11. Schön DA, Wiggins G (1992) Kinds of seeing and their functions in designing. Design Studies 13: 135-156
12. Suwa M, Tversky B (1997) What do architects and students perceive in their design sketches? A protocol analysis. Design Studies 18: 385-403
13. Schön DA (1988) Designing: Rules, types and worlds. Design Studies 9: 181-190
14. Gentner D, Bowdle B, Wolff P, Boronat C (2001) Metaphor is like analogy. In D Gentner, KJ Holyoak and BN Kokinov (eds) The analogical mind: Perspectives from cognitive science (pp. 199-253). MIT Press
15. Casakin HP (2007) Factors of metaphors in design problem-solving: Implications for design creativity. International Journal of Design 1: 21-33
16. Stones C, Cassidy T (2010) Seeing and discovering: how do student designers reinterpret sketches and digital marks during graphical design ideation? Design Studies 31: 439-460
17. Lawson B, Loke SM (1997) Computers words and pictures. Design Studies 18: 171-183
18. Dreyfus HL (1992) What computers still can't do: a critique of artificial reason. MIT Press
19. Setchi R, Bouchard C (2010) In search of design inspiration: a semantic-based approach. Journal of Computing and Information Science in Design 10: 40-47
20. Stiny G (2006) Shape: Talking about Seeing and Doing. MIT Press
21. Mitchell WJ (2001) Vitruvius redux. In EK Antonsson, J Cagan, (eds) Formal Engineering Design Synthesis (pp. 1-19). Cambridge University Press
22. Saund E, Moran T (1994) A perceptually supported sketch editor. In Symposium on User Interface Software and Technology (pp. 175–184). ACM
23. Jowers I, Prats M, McKay A, Garner S (2011) Design exploration with useless rules and eye tracking. In SJ Culley, BJ Hicks, TC McAloone, TJ Howard, A Dong (eds) ICED11 - the 18th International Conference on Engineering Design (pp. 443-455). The Design Society
24. Bucciarelli L (1994) Designing engineers. MIT Press
25. Ferguson ES (1994) Engineering and the mind's eye. MIT Press

26. Henderson K (1999) On line and on paper: visual representations, visual culture and computer graphics in design engineering. MIT Press
27. van der Lugt R (2005) How Sketching Can Affect the Idea Generation Process in Design Group Meetings. Design Studies 26: 101-122
28. Nielson I, Lee J (1994) Conversations with graphics: implications for the design of natural language/graphics interfaces. International Journal of Human-Computer Studies 40: 509-541
29. Maier AM, Kreimeyer M, Lindemann U, Clarkson PJ (2009) Reflecting communication: a key factor for successful collaboration between embodiment design and simulation. Journal of Engineering Design 20: 265-287
30. Shah JJ (1998) Experimental investigation of progressive idea generation techniques in engineering design. In Proceedings of ASME Design Engineering Technical Conference
31. Stacey M, Eckert C (2003) Against Ambiguity. Computer Supported Cooperative Work 12:153-183
32. Tang JC (1991) Findings from observational studies of collaborative work. International Journal of Man-Machine Studies 34: 143-160
33. Lang SYT, Dickenson J, Buchal RO (2002) Cognitive factors in distributed design. Computers in Industry 48: 89-98
34. Gu N, Kim MJ, Maher ML (2011) Technological advancements in synchronous collaboration: the effect of 3D virtual worlds and tangible user interfaces on architectural design. Automation in Construction 20: 270-278
35. Maher ML, Bilda Z, Gül LF (2006) Impact of collaborative virtual environemtns on design behaviour. In JS Gero (ed) Design Computing and Cognition '06, (pp. 305-321). Springer
36. Shapiro V, Tsukanov I, Grishin A (2011) Geometric Issues in Computer Aided Design/Computer Aided Engineering Integration. Journal of Computing and Information Science in Engineering 11
37. Pelechano N, Malkawi A (2008) Evacuation simulation models: Challenges in modeling high rise building evacuation with cellular automata approaches. Automation in Construction 17: 377-385
38. Li AI-K (2004) Styles, grammars, authors and users. In JS Gero (ed) Design Computing and Cognition '04 (pp. 197-215). Kluwer Academic Publishers
39. Mengoni M, Colaiocco B, Peruzzini M, Germani M (2011) Design of a tactile display to support materials perception in virtual environments. In IEEE Virtual Reality 2011 (pp. 227-228)
40. Cagan J, Mitchell WJ (1993) Optimally directed shape generation by shape annealing. Environment and Planning B: Planning and Design 20: 5-12
41. Gupta SK, Nau DS, Regli WC (1998) IMACS: A case study in real-world planning. Intelligent Systems and their Applications 13: 49-60
42. Alink T, Eckert C, Ruckpaul A, Albers A (2010) Different function breakdowns for one existing product: experimental results. In JS Gero (ed) Design Computing and Cognition '10, (pp. 405-424). Springer
43. Fischer A (2011) Engineering-orineted geometry methods for modelling and analyzing scanned data. Journal of Computing and Information Science in Engineering 11

44. Regli W, Kopena J (2010) Challenges in semantics for computer-aided designs: a position paper. In AAAI Cognitive Shape Processing
45. Venkataraman S, Shah JJ, Summers JD (2001) An investigation of integrating design by features and feature recognition. In IFIP Conference, FEATS
46. Bidarra R, Bronsvoort WF (2000) Semantic feature modelling. Computer-Aided Design 32: 201-225
47. Shah JJ, Anderson D, Kim YS, Joshi S (2001) A discourse on geometric feature recognition from CAD models. Journal of Computing and Information Science in Engineering 1: 41-51
48. van den Berg E, Bronsvorrt WF, Vergeest JSM (2002) Freeform feature modelling: concepts and prospects. Computers in Industry 49: 217-233
49. Bronsvoort WF, Noort A (2004) Multiple-view feature modelling for integral product development. Computer-Aided Design 36: 929-946
50. Jha K, Gurumoorthy B (2000) Automatic propagation of feature modification across domains. Computer-Aided Design 32:691-706
51. da Silveira G, Borenstein D, Fogliatto FS (2001) Mass customization: Literature review and research directions. International Journal of Production Economics 72: 1-13
52. Shea K, Ertelt C, Gmeiner T, Ameri F (2010) Design-to-fabrication automation for the cognitive machine shop. Advanced Engineering Informatics, 24: 251-268
53. Bourne D, Corney J, Gupta SK (2011) Recent advances and future challenges in automated manufacturing planning. Journal of Computing and Information Science in Engineering 11
54. Ahn et al. (2001) CyberCut: An internet-based CAD/CAM system. Journal of Computing and Information Science in Engineering 1: 52-59
55. Yan X, Gu P (1996) A review of rapid prototyping technologies and systems. Computer-Aided Design 28: 307-318
56. Paterson G, Earl C (2010) Line and Plane to Solid: Analyzing Their Use in Design Practice through Shape Rules. In JS Gero (ed) Design Computing and Cognition ’10, (pp. 251-267). Springer
57. Searle J (1980) Minds, Brains and Programs. Behavioral and Brain Sciences 3: 417-457
58. Salustri FA (2002) Mereotopology for product modelling: a new framework for product modelling based on logic. Journal of Design Research 2(1)
59. Whitehead AN (1929) Process and Reality. Cambridge University Press
60. Clarke B (1985) Individuals and Points. Notre Dame Journal of Formal Logic 26: 61-75
61. Gerla G (1995) Pointless Geometries. In F Buekenhout, W Kantor (eds) Handbook of Incidence Geometry (pp. 1015-1031). North-Holland
62. Hanna S (2010) Design agents and the need for high-dimensional perception. In JS Gero (ed) Design Computing and Cognition ’10, (pp. 115-134). Springer
63. Stouffs R, Krishnamurti R (2004) Data views, data recognition, design queries and design rules. In JS Gero (ed) Design Computing and Cognition ’04 (pp. 219-238). Kluwer Academic Publishers