Design Intent in 2D CAD: Definition and Survey

Ganeshram R. Iyer and John J. Mills

University of Texas at Arlington, ganeshramiyer@gmail.com, mills@mae.uta.edu

ABSTRACT

Design intent has received much attention in numerous disciplines ranging from mechanical engineering design, software engineering to human-computer interaction research. While it is universally acknowledged that knowing design intent is extremely useful, there is still a lack of both support for frameworks for design intent and widespread use in engineering. Design groups looking to use the latest design intent systems and tools face the problem that the bulk of their product data resides in unintelligent 2D drawings. There exists a need to propagate the design intent present in these 2D drawings to a structured, intelligent, reusable format. This paper tries to understand what design intent is in the domain of 2D CAD drawings by providing a definition for it along with a survey of research and literature in the area of design intent. The paper also focuses on the importance, capture, representation and retrieval of design intent from 2D CAD drawing.

Keywords: design intent, rationale, 2D drawings, context, model.

1. INTRODUCTION

The use of CAD/CAE in design documentation and modelling is becoming ubiquitous [1]. Feature-based CAD systems have demonstrated clear potential for creating attractive design environments and facilitating geometric reasoning related to design function, performance evaluation, manufacturing process planning, NC programming and other engineering tasks. In the last decade, interest in design intent systems has grown. Design intent systems are important tools because they can include not only the reasons behind a design decision but also the justification for it, the other alternatives considered, the tradeoffs evaluated, and the argumentation that led to the decision. The use of a design intent system - a tool for capturing and making design intent easily accessible - can thus improve dependency management, collaboration, reuse, maintenance, learning, and documentation. On the down side, while it was expected that solid modelling or design intent systems would replace drafting systems in design, this turned out to not be the case. Even today, most CAD applications are based on two-dimensional drafting. Shah [2] states that the reason for this failure is the deficiency of the geometric modelling tools. As the design and the manufacturing process evolve around the geometric shape of the product, the current generation of CAD systems is based on geometric modelling techniques. These techniques have proved to be deficient as their usefulness is limited to recording the embodiment detail of the product. Unfortunately designers no longer merely exchange geometric data but need to share more general information about the product such as the design intent, constraints, specifications and manufacturing knowledge. As design becomes increasingly knowledge intensive, the need for computational frameworks to effectively support the formal representation, capture, retrieval and reuse of product knowledge/design intent, becomes more critical [3]. Commercial and governmental entities looking to use design intent systems to improve their product development process, have to deal with the bulk of the design intent that resides in their current design data, such as the 2D drawings. This design intent needs to be propagated to a more reusable, intelligent and structured format such as those used by design intent or knowledge-based systems.

The purpose of this paper is to help understand design intent in 2D CAD, its significance and discuss the open issues present in the capture, representation and retrieval of design intent. We do so by providing a definition for design intent in this limited domain as well as a survey of research and literature. The paper is organized into the following sections: 1) literature survey with a focus on defining design intent in the generic sense as well as in the limited 2D CAD domain, 2) importance of design intent and disagreement in its interpretation and 3) discussion of open issues.

2. LITERATURE SURVEY

There are limits to the scope and nature of this review. We have largely reviewed papers in the 2D CAD area that describe implementation specific details, some of which are limited to mechanical engineering drawings. Section 2.1

summarizes research in the area of generic design intent. The survey of design intent is focused on determining its definition, importance and the consensus, or lack thereof, among various researchers and relevant literature about its contents. It is by no means intended to be as comprehensive a review as Hu et al's survey paper [4] on capture, representation and retrieval frameworks for design rationale. With our survey we are looking to contribute a definition for design intent, which was missing from [4]. Section 2.2 summarizes research in the area of design intent in 2D CAD. A brief explanation is provided to justify the need to capture design intent from 2D CAD along with the failings of current research to do so.

2.1 Design Intent

The interpretations of design intent in general range from treating it as a historical record of analyses and decisions that led to the choice of the particular artifact or feature in question [7] to treating it as the sum of the features (functional, geometric, constraint etc.) and the attributes of the features.

Conklin and Yakemovic [5] consider design rationale as the path of decision and selected alternatives that join the initial state to the final state. They define design rationale as the reason an artifact is structured the way it is and has the behaviour that it does. They state that the capture of design rationale enhances the design process, which is necessary as humans poorly manage exploration of the space of design decisions, especially on large, complex development projects. The paper gives details about the gIBIS software tool based on the Rittel's Issue-Based Information Systems [IBIS] which attempts to capture design rationale non-intrusively.

Fischer et al. [6] point out that documenting design rationale can support maintenance and redesign of an artifact and reuse of the design knowledge. To alter a design sensibly – adding, fixing or modifying features – it is crucial to have an understanding of why it was designed in a particular way. Thus, documenting design rationale has great potential for improving design and cooperation and understanding existing artefacts. Fischer et al's paper approaches design rationale as statements of the reasoning underlying the design process that explain, derive and justify design decisions. To this purpose of documenting design rationale, they present the evolution of the conceptual framework [Procedural Hierarchy of Issues – PHI], developed to integrate design environments and document rationale. They do acknowledge that a truly complete account of the reasoning relevant to design decisions is not possible due to their implicit nature and human behaviour.

J. Lee and K. Lai [7] define design intent or rationale as an explanation of why an artifact or some part of it is designed the way it is. They built a framework to support a number of design tasks and use that framework to present and assess a language for representing rationales, termed the Decision Representation Language. The article points out that an explicit representation of design rationale has many benefits. Such a representation can lead to a better understanding of the issues involved and of the design space. The article states that design rationale includes all the background knowledge such as deliberating, reasoning, trade-off and decision-making in the design process of an artifact—information that can be valuable, even critical, to various people who deal with the artifact. Lee and Lai further use the term design rationale in at least three different ways: a historical record of the reasons for the choice of an artifact, a set of psychological claims embodied by an artifact and description of the design space. The tasks that a design rationale/intent representation can or should support can be described in many ways at different level of abstraction e.g. documentation, understanding, debugging, verification, analysis, explanation, modification and automation.

Henderson [8] has a similar understanding of design intent: that it is the underlying purpose behind an object. He differentiates between functionality and intent in that intent justifies a design decision whereas functionality just tells what the design does. The author asserts that current CAD tools provide limited assistance in situations requiring the designer or the re-designer to query the rationale behind a design decision. While the CAD tools capture the shape of each of the components and annotate it they do not capture the functionality or the design intent of the products. The author presents a prototype system that captures the meta-semantics of entities including dimensions and tolerances along with the features and functionality. The model structure is based on the concept of a Product Definition Unit (PDU) as a basic element that can represent anything from an entire system to a single feature. Design intent is viewed as a characteristic of a PDU. The PDUs are defined in a particular context which includes a level of abstraction, a level of detail, a specific design objective, a number of PDUs and the relations and constraints among them. Design intent subsets such as relations and alternatives are represented as instances of PDUs in a particular context.

Hounsell [9] and Case make a case for the importance of design intent by stating its need in validation systems capable of reasoning about the semantics in a particular design. They interpret the designer's intent as the sum of volumetric, morphological, semantic, functional, and validation features and the relationships between the features themselves or elements of the features. They claim features to be carriers of designer's intent and make a case for their inclusion in a feature-based modelling system. They assert that conventional CAD systems are incapable of capturing non-geometric aspects of design intent such as design alternatives, design procedures, functionality etc, which act as a bridge between this set of abstract objects and the spatial model consisting of geometry, material, dimensions, surface finish etc. The authors briefly describe a prototype system called FRIEND (Feature-based Reasoning system for Intent-driven ENgineering Design) that more clearly defines a feature's semantics using the design intent presented.

Mills and Wang [10] define intent/rationale as the justification of design decisions in terms of selecting values for structure variables to satisfy constraints. This definition recognizes design intent as the justification for the physical values attributed to artifact structure thereby merging the generic view of design intent with the domain specific view from architecture, engineering etc.

Horvath and Rudas [11] define design intent as the emergence of some form of application and domain specific knowledge. They categorize design intent into three levels viz. application, relationship and representation with corresponding attributes including type of intent, status of intent and status of decision maker. The type of intent is similar to purpose and is further categorized into i) possible alternatives, ii) compatibility, iii) fixed characteristics, iv) results of tests, v) application type, vi) intended strategy, vii) counter proposal viii) allowable range, ix) pros and contras. They state also that design intent is the intellectual arrangement of features and dimensions of a design i.e. if one were to know the relationships and dimensional variables then a significant portion of design intent is known.

Another popular interpretation of design intent is as the sum of the function, behaviour and structure of the design artifact. Rosenman and Gero [1] [12], Gero and Kannengiesser [13] and Fenves et al [14] all propose frameworks that account for design knowledge using those three variables linked by processes. Rosenman and Gero mainly deal with representation of design intent and argue for a multi-view approach for any meaningful representation in a multidisciplinary environment. They present the paradigm of purpose-function-behaviour-structure to describe the product engineering semantics. Gero and Kannengiesser [13] [15] describe a process and representation framework that relies on artificial intelligence and design agents to develop a design model that does not require the design intent to be encoded a priori. Fenves et al [14] propose a conceptual data architecture that can provide the technical basis for making tighter integration of spatial and functional design, analysis-driven design and, eventually, opportunistic analysis more pervasive. They propose a representation model (a master model comprising function, behaviour and structure) that integrates with a function-driven design scenario. While the models proposed by [12], [13], [14] and [15] do not directly attempt to define or represent design intent they account for design decisions and design variables that record history of model process and design knowledge, all important connection points for intent description.

Nielsen, Dixon and Zinsmeister [16] present a design-by-features system in which the features are defined as processing not only form but also certain designer's intentions regarding geometric relationships. In contrast to a constraint-driven simultaneous equation solving methods, this system uses an intent-driven knowledge based method to represent design process. Geometric intent is modelled as the collection of all restraints (scalar limit values or target values) on geometric attributes of the designer's form.

Brissaud, Garro and Poveda [17] represent design intent as design alternatives, decision-making and design constraints. They describe an approach that captures the intent of a product with proposed conjectures and conjecture evaluation criteria as elements. The origins of the product solution finally adopted may be retraced through the history of the criteria and conjectures processed during the preceding design process. Hayes, Sevy and Regli [18] also describe a method that captures and represents important aspects of design intent, the evolution of the solid model through the lifecycle and the process history. While [18] only talks about capturing and representing design intent as evolving geometric information, the method deals with conflict resolution and change management, which is important to maintain consistency in the captured design intent.

2.2 Design Intent in 2D CAD

Most definitions and interpretations discussed in section 2.1 serve to provide a generic view of design intent. Most attempts to merge geometric, volumetric and structural information with design intent occur in the 3D domain [9] [18] but none do so comprehensively. The merging of the geometric information in 3D is done at the feature level and is usually done during the design process. This eliminates the possibility of using existing knowledge in the form of 2D drawings and 3D models, the current storehouses of design intent and manufacturing knowledge, to capture and augment design intent. It should be noted, however, that the design intent that can be captured from the 2D drawings will be limited due to the nature of the information present on it. 2D drawings contain only unstructured graphic entities such as lines, texts and symbols. Humans add the semantics to interpret these graphic entities as a coherent representation of some design object [19]. An approach to identify these semantics has been lacking.

There exists much research to capture the information present on the 2D drawings and propagate it to an intelligent, parametric, feature-based 3D model. Weiss and Dori [20] propose an approach that automates the 3D object reconstruction from 2D engineering drawings by mimicking human intelligence. Combining elements of variational geometry, matrix algebra and graph theoretic methods the approach incorporates high level understanding of 2D engineering drawings, topological relations and dimensional scheme analysis for each 2D view. To extract the high level information from the 2D drawing it is viewed as a dual faceted entity composed of several layers. Given an orthographic view on the 2D drawing they automatically extract constraints which describe implicit and explicit dimensions using variational geometry, based on translating each constraint to an algebraic equation. The 3D reconstruction is done by merging the dimensioning scheme of each 2D view into a common dimensioning scheme for the entire object, generating a composite network, labelling the dimensions in 3D and converting the network to the object. Dori and Wenyin [21] have described a complete system that realizes the entire process of understanding mechanical engineering drawings from scanning to 3D reconstruction. The system described has the capability of separating geometric entities from non-geometric, such as text, arrowheads, leaders, dashed lines and hatch lines etc.

Tanaka et al [22] describe a method to automatically convert 2D assembly drawings to 3D part drawings, generating a unique solution for designers regardless of the complexity of the original 2D assembly drawings. They use the dimension lines, part numbers and part lists, usually drawn on the 2D, to do so. The algorithm accepts as input 2D assembly drawings with non-complex surfaces, separates the assembly into virtual 2D part drawings, recognizes blocks from the part drawings, searches for true blocks for each part and finally outputs a 3D assembly drawing.

Joseph [23] has presented a methodology for the interpretation of engineering drawings based on a combination of schemata describing drawing constructs with a library of low-level image analysis routines and a set of explicit control rules applied by a parser. The resulting system integrates bottom-up and top-down processing strategies within a single, flexible framework modelled on the human perception cycle. The system, termed Anon, is a knowledge based image analysis system intended to extract 2D graphical elements and symbols from a grey level image of a mechanical engineering drawing. The system classifies the information on the drawing into appropriate schematic classes such as solid, dashed and chained lines, solid and dashed curves, cross hatching, text, witness and leader lines and certain forms of dimensioning. All of Anon's image analysis is carried out under the control of some given schema – in the context of a particular hypothesis regarding the local content of the drawing.

Cheng and Yang [24] propose a knowledge-based graphic description tool that is used to recognize and understand engineering drawings. The graphic description tool basically consists of a concept description network, a graphic description language, a physical description framework, a set of image processing modules, a matcher, a rule-based inference engine, an interpreter and blackboard control architecture. The concept description framework, graphic description language, and physical description framework are designed to represent domain knowledge, graphic semantic knowledge and physical properties of engineering drawings in different fields. The matcher recognizes all graphic symbols and characters that are extracted by the low-level image processing routines. The rule-based inference engine is built to infer possible relations among graphic symbols and generate a relational graph. The interpreter is used to generate an acceptable explanation in terms of traversal of the relational graph. This framework does not attempt to create a solid model from the captured information but instead builds an engineering drawing understanding system that could be queried as necessary.

Vaxiviere and Tombre [25] present a knowledge based system named CELESSTIN that extracts technologically significant entities and analyzes the whole setup with respect to disassembling and kinematics knowledge. These technologically significant entities allow CELESSTIN to start using rules referring to the semantics of the represented object itself. The paper illustrates how to assign a simple syntax on the basic structures to recognize simple mechanical entities such as shafts or screws. Then semantic knowledge is introduced in the reasoning. The multi-expert system disassembles the whole mechanical setup and analyzes the kinematics of the represented device thus inferring about the functionalities of the different mechanical entities. This systematic rules propagation leads to the identification of mechanical entities such as shafts, screws etc.

Various other research attempt to solve this problem and can be found in [26], [27], [28], [29]. Almost all solutions, excluding Cheng's and Yang's, and Vaxiviere's and Tombre's, address the problem of 2D drawings to 3D model conversion as mostly geometric with possible input from the supporting symbols and text that may be present on the 2D drawing sheet. Research to identify and separate product geometry from dimension sets, arrowheads, hatching lines, text and symbols fall short in failing to recognize that the non-geometric information on the drawing sheet contributes to engineering knowledge, design intelligence and some design intent. It is the contention of our present contribution that these are the aspects that play a major role in design intent capture. Tanaka's solution is further limited, in that one major requirement for their algorithm to work is that the original assembly drawings consist of standard parts such as bars and plates. While Cheng and Yang's [24] paper describes a rule based system that recognizes, examines and classifies the graphic symbols in the engineering drawings, their graphic description language diverges from the current practice of using vectorized geometric information. The specific domain knowledge of the drawing that their system extracts is mostly used to examine and classify the graphical symbols in a field. Very little semantic knowledge is attached to the graphical symbols using the domain knowledge. Vaxiviere's and Tombre's CELESSTIN [25], while able to recognize simple mechanical entities, will face difficulties when the complexity of the mechanical entities grows beyond symmetric blocks. CELESSTIN also lacks support for non-geometric entities, which could pose problems when information is missing from the geometry of the artefact.

3. DESIGN INTENT: IMPORTANCE, CONSENSUS AND SUGGESTED DEFINITION

3.1 Importance and Consensus

Design intent is considered important for various reasons. Pena-Mora et al. [30], while they do not attempt to define design intent, state that the Architecture/ Engineering/Construction industries can benefit from the explicit representation of the design process rationale in many ways;

(i) Large and lengthy projects change over time and require certain design decisions to be modified during the design-construction process. Reasons or justifications used during the initial design stages can be lost resulting in the need to define them over and over resulting in increased project costs and delays. The ability to store and recall these reasons will improve productivity.
(ii) The quality of the project increases as the project intents are represented explicitly and are readily accessible for review.
(iii) A model that allows the intent to be explicitly stated and easily manipulated leads to a more intelligent use of knowledge and resources.
(iv) Understanding design intent of designers is also important to achieve coherent integration of design solutions and transfer design knowledge [31].

With the definitions and interpretations presented in section 2.1, 2.2 and the summary above, it is clear that while there is consensus on the importance of design intent, there is little agreement on its exact meaning. There are some portions of the definition that are common to all interpretations. Some of these are; design intent

(i) implies more than just geometry,
(ii) justifies decision among alternatives
(iii) serves as a historical record of analyses and processes
(iv) includes variables such as function, behaviour, volumetric etc
(v) is application and domain specific
(vi) evolves throughout the development lifecycle
(vii) describes design space etc.
(viii) depends on the context of the current problem

The other portions e.g. the representation of evolving geometric information, while relevant are not common interpretations. We perceive the reasons for the lack of consensus to be a lack of commonality in the context in which the interpretation is made. The individual interpretations have been made in varying domains considering numerous, though limited, applications.

3.2 Suggested Definitions for Design Intent

The requirements and interpretations stated in section 2.1 form the basis of our suggested definition for design intent: *"Design intent is application, domain and context dependent knowledge that describes design space, represents design alternatives and process history, justifies design solutions and decisions and determines the characteristics of features and entities and the relationships among them"*

This definition can be dissected and the relevance of each individual sub-part can be traced back to the discussion provided in section 2. The definition provided above while useful when considering the generic view of design proves too vague or unhelpful when dealing with the limited domain of 2D CAD drawings. Using this generic definition of design intent we can further suggest a definition for design intent in the domain of 2D CAD: *"Design intent contained in legacy CAD is the insight into the design variables (design objectives, constraints, alternatives, evolution, guidelines, manufacturing instructions and standards) implicit in the structural, semantic and practical relationships between the geometric, material, dimensional and textual entities present in the CAD representation"*

This is a subset of our suggested, broader definition of design intent. The context for the second definition is the 2D CAD domain along with the domain the drawing belongs to viz. mechanical, electrical, architecture etc. Fig. 1 uses an illustration to depict the classification of the constituents of design intent in legacy CAD. It should be noted that this illustration does not show the intricate relationships that exist between the sub-divisions of syntax, semantic and pragmatic, which is important when describing design intent. We show these relationships in Figure 2. The entities contained on the CAD representation can be organized using three levels viz. structural (syntactic), semantic and practical (pragmatic). Design intent can be inferred from the associations and relationships that exist between the entities contained on the CAD representation. We present elsewhere an inference approach using this idea to capture design intent [32].

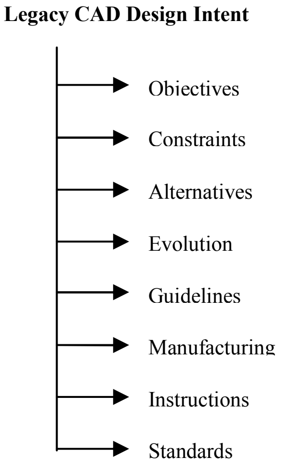

Fig. 1. Illustration of Design Intent in Legacy CAD.

4. DISCUSSION

We have discussed design intent/rationale from two perspectives: the general view of design intent and the view of design intent in the context of 2D drawings. With this paper we have addressed one open issue in the area of design intent in 2D drawings i.e. the definition of design intent. Many other issues still remain open and we discuss them in this section.

4.1 Open Issues

4.1.1 Capture of design intent on 2D drawings

The nature of the systems that capture generic design intent are traditionally either process oriented, which emphasize design intent as a history of the design process, or feature oriented, which support domain knowledge-bases [4]. One of the problems that need to be addressed is that the approach to automate design intent capture from 2D drawings does not fit well into either of these approaches. The process oriented approach starts at the very moment conceptual design begins. When dealing with 2D drawings the design artifact is already in the manufacturing/maintenance of its lifecycle and the history of the process is largely unavailable. This hinders the use of the process oriented approach. The feature-oriented approach would only work if the intent captured from 2D drawings provided sufficient information to validate against a knowledgebase. This poses some problems as the completeness of design intent that one is able to capture from 2D drawings is as yet unknown. Determining the nature of any new approach to capture design intent from 2D drawings is the primary issue that needs to be addressed. Further research and development in this area could contribute better tools to that extent.

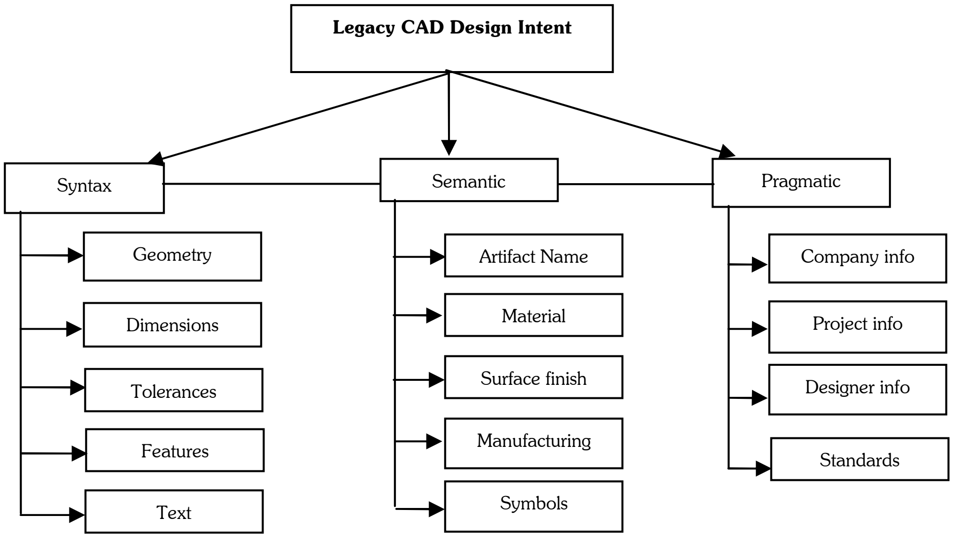

Fig. 2. Design intent represented by syntax, semantic and pragmatic relationships.

4.1.2 Representation of design intent capture from 2D drawings

Hu et al. [4] state that there is a considerable amount of work needed in design intent representation [4]. They clarify that one of the issues in design intent systems should be the capability to represent potentially relevant features and combine feature of objects in specific contexts to form coherent explanations. The approaches reviewed in section 2.2 deal mostly with the process of capture and propose little in the area of representation of the design intent captured from the 2D drawing. While generic design intent representation has received much attention, little research has been performed in the area of design intent on 2D drawings. A survey of the representation schemas proposed for generic design intent is required with the objective of determining their suitability for representing the same in the context of 2D drawings. Alternatively other comprehensive representation schemas, which address representation of complete product knowledge, could be considered. Some examples of alternate schemas are the STandard for the Exchange of Product data (ISO/STEP) and NIST's Core Product Model [33] (CPM) and Open Assembly Model [34] (OAM). The Core Product Model and the Open Assembly Model are especially suited for the task of design intent representation as they contain classes that explicitly include design rationale.

4.1.3 Retrieval of design intent captured from 2D drawings

In section 1 we briefly discussed the yet to be realized need to propagate the design intent present in 2D drawings to a more reusable format. Hu et al. [4] state that another important issue in design intent systems is to encode the modelling knowledge in a form that can be shared and reused by several applications. The ability to retrieve the design intent that has been captured from 2D drawings and stored a suitable representation schema has gone unaddressed in almost all of the literature surveyed in this paper. While capture and representation are the pressing issues for current research much effort is needed that addresses the open issue of retrieval of design intent captured from 2D drawings.

4.1.4 Domain aspects of design intent

The literature surveyed tends to propose an approach for capture of design intent from 2D drawings in the mechanical engineering domain. While this domain is a good candidate to prove the proposed approaches it does not adequately represent the sample space of 2D drawings. 2D drawings are widely used in various domains ranging from architecture to ship building. The papers by Cheng and Yang [24] did explicitly consider domain knowledge-bases and we acknowledge their approach, while vastly limited, as a step in the right direction.

5. CONCLUSION

With this survey we have covered a number of design intent systems in the context of 2D drawings and suggested a definition for the generic view of design intent and a definition for design intent in the context of 2D drawings. We hope that we have contributed to the understanding of the issues still open in this area while suggesting new avenues for future research.

6. REFERENCES

[1] Rosenman, M. A. and Gero, J. S, Purpose and function in a collaborative CAD environment, *Reliability Engineering and System Safety*, Vol. 64, 1999, pp 164-179.

[2] Shah, J. J. and Mäntylä, M., *Parametric and Feature-based CAD/CAM: Concepts, Techniques and Applications*, ISBN 0-471-00214-3, 1995.

[3] Szykman, et. al., The Role of Knowledge in Next-generation Product Development Systems, *ASME Journal of Computing and Information Science in Engineering*, 2001.

[4] Hu, X., Pang, J., Pang, Y., Atwood, M., Sun, W. and Regli, W., A Survey on design rationale: Representation, Capture and Retrieval, Proceedings of DETC'00: 2000 ASME Design Engineering Technical Conferences, September 10-13, 2000

[5] Conklin, E. J. and Yakemovic, K. C., A process oriented approach to design rationale, *Human-Computer Interaction*, Vol. 6, 1991, pp 357-391.

[6] Fischer, G., Lemke, A. McCall, C. R. and Morch, A. I., Making argumentation serve design, *Human-Computer Interaction*, Vol. 6, No. 3-4, 1991, pp 393-419.

[7] Lee, J. and Lai, K., What's in design rationale, *Human-Computer Interaction*, Vol. 6 Nos. 3-4, 1991, pp 251–280.

[8] Henderson, M. R., Representing Functionality and Design Intent in Product Models, in 2nd ACM Solid Modeling 1993, Montreal, Canada.

[9] Hounsell, M. S., and Case, K., Morphological and volumetrical feature-based designer's intents. In Advances in Manufacturing Technology XI, The Proceedings of the 13th National Conference on Manufacturing Research. 1997. Glasgow, Scotland.

[10] Wang, F. and Mills, J., A Product Data Model and Processor for Supporting Virtual Product Development. in Proceedings of the DETC: ASME 2000 Computers and Information in Engineering Conference.

[11] Horvath, L. and Rudas, I. J., Modeling Behavior of Engineering objects using design intent model, Industrial Electronics Society, IEEE, 2003.

[12] Rosenman, M. A. and Gero, J. S., CAD modeling in multidisciplinary design domains, *Artificial Intelligence in Structural Engineering*, 1998, pp 335-347.

[13] Gero, J. S. and Kannengiesser, U., Function-Behavior-Structure: A Model for Social Situated Agents. 2003.

[14] Fenves, S., Choi, Y., Gurumurthy, B., Mocko, G. and Sriram, R., Master Product Model for the Support of Tighter Integration of Spatial and Functional Design, NIST, 2003.

[15] Gero, J. S. and Kannengiesser, U., The situated function-behavior-structure framework, in J. S. Gero (ed.), Artificial Intelligence in Design '02, Kluwer, Dordrecht, 2002, pp 89-104.

[16] Nielsen, E., Dixon, J. and Zinsmeister, G., Capturing and using Designer Intent in a Design-With-Features System, DE-Vol 31, Design Theory and Methodology, ASME 1991.

[17] Brissaud, D., Garro, O. and Poveda, O., Design process rationale capture and support by abstraction of criteria, *Research in Engineering Design*, Vol. 14, 2003, pp 162-172.

[18] Hayes, E. E., Sevy, J. and Regli, W. C., Representation of Temporal Change in Solid Models, 6th ACM/SIGGRAPH Symposium. on Solid Modelling and Applications, Ann Arbor, MI, 2001, pp 317-318.

[19] Rosenman, M. A. and Gero, J. S., Purpose and function in a collaborative CAD environment, *Reliability Engineering and System Safety*, Vol. 64, 1999, pp 164-179.

[20] Weiss, M. and Dori, D., A Scheme for 3D Object Reconstruction from Dimensioned Orthographic Views. IEEE, 1995, pp 335-338.

[21] Dori, D. and Wenyin, L., Automated CAD Conversion with the Machine Drawing Understanding System: Concepts, Algorithms and Performance. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, Vol. 1, No. 4, 1999, pp 411-416.

[22] Tanaka, M., A single solution method for converting 2D assembly drawings to 3D part drawings, *Computer Aided Design*, Vol. 36, No. 8, 2004, pp 723-734.

[23] Joseph, S. H. and Pridmore, T. P., Knowledge-directed interpretation of mechanical engineering drawings, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 9, 1992, pp 928-940.

[24] Cheng, Y. Q. and Yang, J. Y., A Knowledge-based Graphic Description Tool for Understanding Engineering Drawings, Proceedings of the 2nd International IEEE Conference. 1990.

[25] Vaxiviere, P. and Tombre, K., CELESSTIN IV: knowledge-based analysis of mechanical engineering drawings, IEEE International Conference. 1992.

[26] Devaux, P. M. et al, A Complete System for Recovery of 3D Shapes from Engineering Drawings, IEEE, 1995, pp 145-150.

[27] Lai, C. P. and Kasturi, R., Detection of Dimension Sets in Engineering Drawings, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 8, 1994, pp 848-855.

[28] Vaxiviere, P. and Tombre, K., Celesstin: CAD Conversion of Mechanical Drawings, IEEE COMPUTER Magazine, Vol. 25, No. 7, pp 46–54.

[29] Chen, K.-Z. and Feng, X. A., Solid model reconstruction from engineering paper drawings using Genetic Algorithms. *Computer-Aided Design*, Vol. 35, No. 13, 2003, pp 1235-1248.

[30] Pena-Mora, F., Sriram, D. and Logcher, R., SHARED-DRIMS: SHARED design recommendation-intent management system, Enabling Technologies: Infrastructure for Collaborative Enterprises, Second Workshop on 20-22 April 1993, pp 213 – 221.

[31] Ishino, Y., (IMPACT Laboratory, Dept of Aerospace Engineering, Univ. of Southern California), Y. Jin, Source: *Advanced Engineering Informatics*, Vol. 16, No. 2, 2002, pp 107-125.

[32] Iyer, G. R., Mills, J. J., Barber, S., Devarajan, V. and Maitra, S., Using a Context-Based Inference Approach to Capture Design Intent from Legacy CAD, *Computer-Aided Design and Applications*, Vol. 3, Nos. 1-4, 2006, pp 269-278.

[33] Fenves, S. J., A Core Product Model For Representing Design Information, National Institute of Standards and Technology, NISTIR6736, Gaithersburg, MD, 20899, USA, 2001.

[34] Baysal, M. M., Roy, U., Sudarsan, R., Sriram, R. D. and Lyons, K., The Open Assembly Model for the Exchange of Assembly and Tolerance Information: Overview and Example, Proceedings of the DETC 2004 ASME Design Engineering Technical Conferences, Salt Lake City, UT, September 28 - October 2, 2004.