Computers in Industry 65 (2014) 1041–1052

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

Ontology based automatic feature recognition framework

Qingmai Wang *, Xinghuo Yu

School of Electrical and Computer Engineering, RMIT University, GPO Box 2476, Melbourne, VIC 3001, Australia

ARTICLE INFO

Article history:
Received 25 October 2013
Received in revised form 8 April 2014
Accepted 9 April 2014
Available online 27 May 2014

Keywords:
Automatic feature recognition
Ontology
Computer-aided design

ABSTRACT

AFR has long been realized as a key technology for design automation. A significant shortcoming in AFR is that most of them are individual systems that are isolated from each other, due to the absence of a standard feature library or feature modeling techniques. Few studies attempted to overcome this problem by allowing a certain degree of user customization or extension, which are still far from success. In order to address this issue, this paper proposes an ontology-based feature recognition framework. In the framework, features are captured transparently and hierarchically within a formal OWL ontology, and the feature recognition is achieved by applying an efficient backward-chained ontology reasoner to reason through the ontology. The resulting feature recognition system shows a high level of flexibility, maintainability, and explainability, for both representing and recognizing features. The effectiveness of the framework is finally demonstrated with three case studies.

1. Introduction

Modern shape model design of products or parts heavily rely on computer aided technologies (CAx) including Computer Aided Design (CAD), Computer Aided Manufacturing (CAM) and Computer Aided Process Planning (CAPP). Those systems are becoming highly collaborative and often involve multi-disciplinary project teams at distributed sites with heterogeneous computer systems. To share design information, Standard for the Exchange of Product model data (STEP) is currently recognized as one of the most effective and efficient information exchange methods [1,2]. However, STEP only describes basic geometric information while feature information which captures designer's intent and manufacturing patterns are lost. Since increasingly intelligent CAx requires the geometric model to be interpreted in terms of features [3] and manually feature recognition for growingly complex model with large data volume is impossible, Automated Feature Recognition (AFR) methods are developed. The features here can be explained as abstract concepts regarding some interesting geometrical or topological patterns. The role of AFR systems is to extract such information from basic representation of shape models without interfering users.

AFR has long been realized as a key technology in automated design processes. Comprehensive reviews of existing AFR methods have been provided in [3–5]. According to these reviews, methods that attract more attention can be classified into three categories: graph-based approaches, volumetric decomposition approaches and hint-based approaches.

Graph-based method was firstly proposed in 1988 [6]. The method captures the concave/convex relations of the part's surface within an adjacency graph, and analyze the graph in order to find interesting sub-graphs as features. A number of following techniques, including Multi-Attributed Adjacency Graph (MAAG) [7], Structured Face Adjacency Graph (SFAG) [8], were developed to overcome the difficulties in handling curved faces. The major drawbacks of all graph-based methods are the incapability of handling intersecting features and no guarantee that the recognized feature will prove applicable in the sense of manufacturability [3]. Volumetric decomposition methods decompose the geometric model into sub-volumes repeatedly until the feature is recognized. Convex hull based volumetric decomposition was originally introduced in 1991 as Alternating Sum of Volumes (ASV) decomposition [9]. In ASV decomposition, volumes of input models are decomposed by subtracting them from their convex hull and repeating the process for all the resulting volumes [10]. A critical problem of ASV decomposition is that the algorithm may not converge. To solve this problem, the Alternating Sum of Volumes with Partitioning (ASVP) decomposition was developed by combing ASV and remedial partitioning using cutting operations [11]. Hint-based methods were developed with the motivation of overcoming the difficulty in handling feature interaction. Object-Oriented Feature Finder (OOFF) was firstly proposed in [12], where face patterns in a solid shape's B-rep generate hints or clues for the existence of certain machining features. These hints are tested for

* Corresponding author. Tel.: +61 3 99253616/433970027.
E-mail address: qingmai.wang@rmit.edu.au (Q. Wang).

http://dx.doi.org/10.1016/j.compind.2014.04.004
0166-3615/

Table 1
Difficulties of user customization in existing AFR methods.

AFR methods	Customization
Graph-based methods	Easy for customization as that the features are decoupled and represented as graphs. However, since that the adjacency graphs captures very little information of a solid model, the capability of using the adjacency graphs to represent a feature is quite limited, especially for the interacting features.
Volumetric decomposition	Difficult for customization as that the representation of features is embedded in the recognition process of the features. Therefore the extension of the feature libraries requires extensive modifications for the whole recognition process.
Hint-based approached	Similar with the volumetric decomposition, the feature representation and recognition are integrated together, namely, how a feature can be formed from a hint is also the process how this feature is recognized. As a result, the customization of new features may requires the creation of new hints and new recognition process, which may not even exist.

validity through geometric completion procedures that attempt to construct the largest volumetric feature that is consistent with the boundary data [13]. An advanced system, integrated Incremental Feature Finder (IF^2), is further developed to extend OOFF by providing it with the ability to reason about hints generated from various sources [13].

A conventional benchmark for AFR systems is the capability of handling interacting features. However, it has been recently realized [20] that another major limitation for the existing AFR systems is that they are neither widely applicable nor generalizable. Most of them only work on self-defined feature libraries for isolated applications, with very limited general applicability. An obvious reason for this problem is that building a complete universal feature library is quite difficult, since AFR is essentially a very domain-dependent and application-oriented study. Different AFR systems may have different purposes for recognizing features, some of them focus on generating features for manufacturing purpose [3], others may be targeted in extracting features for engineering analysis [14,15], and, there are also studies simply used for design constraints check [16]. The features to be recognized also have grown from the simple machining features (holes, pockets and slots) to some sophistically complex features (e.g. stepped holes [17], X-junction [18], V-through slot [19], and intersecting grooves [20]) that are purely based on the authors' own preferences, without any standardization, and are not interpretable between each other.

A practical solution to solve the general applicability issue is to develop a method which make the customization and extension of the feature libraries easier and less costly. Unfortunately, the major categories of AFR methods are weak in terms of the capability of allowing user to make customization. Table 1 summarizes their difficulties.

Apart from the major categories, a number of studies were conducted attempting to solve this problem. [21] proposed a high level feature recognition system which enables end-users to define high-level features as combinations of low-level features. Users' script-like descriptions of high-level features are transferred into feature relationship graphs, and subgraph isomorphic techniques are applied for feature recognition. The capability of this system is limited since that it can only process the combinational features and the requirement of extensive user interventions also reduces the level of automatization [5]. Another work that enables feature library extensions is an inductive learning approach that can automatically generate logic rules from sample features as feature hints [22]. However, how to apply these hints to recognize features and how practical this approach could be for complex interacting features are questionable. A more recent study [20] has developed a hybrid procedural and knowledge-based approach based on artificial intelligence planning to address this issue, whereas the absence of formal knowledge representation techniques limit the flexibility and computational efficiency of the system. Besides the research studies, a commercialized software FeatureCAM [23] also supports a way of creating custom features called "User Defined Features". However the major limitation is that it requires the user to have solid Visual Basic programming skill to encode the features.

As a result, in order to be employed in solving AFR tasks in different application domains, this paper proposes a novel ontology-based feature recognition framework. The framework follows the idea of decoupling feature representation and recognition presented in [20], but employs advanced ontological methods to represent the features. The application of ontology in the field of product design is arising in recent years, e.g. [24–26], due to its advantages in sharing information, implementing interoperability, enhancing flexibility and reusing knowledge [27].

An important reason for introducing ontology into AFR is that ontology naturally separate the domain knowledge from the operational knowledge, which exactly fits our intention of decoupling feature representation and recognition if we consider the representation as domain knowledge and the recognition as operational knowledge. Thus the ontology of feature library are quite extensible and scalable since they can grow independently of the recognition process. Another amazing thing of using ontology is the explicit specification of the domain knowledge. Unlike several previous studies which use programming language to encode the features, the ontological explicit representation not only make the features and domain concepts easy to find and understand, but also easy to apply changes. Besides these two reasons, there also a number of additional benefits brought by the employment of ontology techniques, such as: (1) it helps sharing a common understanding of feature representation and feature recognition; (2) it enables re-use of domain knowledge for customizing features; (3) explanation can be easily generated due to the explicit representation of the domain knowledge; and so on.

The rest of the paper is organized as follows: Section 2 gives an overview of the framework; Section 3 discusses the details of the feature ontology; Section 4 describes the feature recognition process; Section 5 presents the case studies; and finally Section 6 concludes the paper.

2. Overview of the framework

The feature recognition framework proposed in this paper is applicable for STEP-based CAD design. STEP is a popular data format for product exchange. The standard represents product information along the necessary mechanisms and definitions, including geometry, topology, tolerance, and etc [5]. Within STEP, the Application Protocol (AP) 203 represents products as explicit non-parametric models based on Boundary Representation (B-rep) [2], which is also the focus of this study.

Fig. 1 shows the architecture of the proposed framework. Based on the idea of decoupling feature representation and recognition, the framework can be generally divided into two major components: a feature ontology which defines features based on the basic STEP AP203 entities; and a feature recognition coordinator which manages the overall feature recognition process and controls the information flow.