

Features Extraction from CAD as a Basis for Assembly Process Planning

Baha Hasan^(✉) and Jan Wikander

Department of Machine Design, The Royal Institute of Technology (KTH),
Stockholm, Sweden
bahasan@kth.se

Abstract. This paper describes a novel approach to recognize product features, which are significant for Assembly Process Planning (APP). The work presented in this paper is a part of a larger effort to develop methods and tools for a more automated and bidirectional link between product CAD and the different processes and resources applied in APP. APP is the phase, in which the required assembly processes and resources are determined in order to convert a product to fully assembled or semi-assembled product. Product features will be extracted from the SolidWorks (SW) CAD file using SW - Application Programming Interface (API). SW-API is an interface that allows the exchange of data between CAD design and different software applications. The work includes automatic recognition for assembly knowledge, geometry and non-geometry knowledge (dimensions, geometrical tolerances, and kinematic constraints) in assembly design, which are relevant for assembly process and resources. Recognition algorithms have been developed by using visual basic. Net (VB.net). A case-study example is included for illustration of the proposed approach.

Keywords: Features · Form features · Assembly Process Planning · CAD · SolidWorks · API

1 Introduction

Assembly is one of the most complex tasks in manufacturing environment; it consumes a considerable percentage of total manufacturing and labor costs [1]. Assembly Process Planning (APP) is an important and decisive link phase between assembly design and manufacturing. The ultimate final aim of APP is to determine the required assembly processes and resources to convert an assembly design into an assembled or semi-assembled product [2]. The integration of product design and assembly process planning (APP) thus has a strong impact on product realization, as also discussed by Du and Zha [3].

Product features have been used by several researchers to improve the efficiency of process planning both in manufacturing and assembly [2, 4–6]. Features are used when designing a product (Feature-based design), while in process planning features are to be extracted from the product model. The extracted features represent a natural link

between product design and process planning domains and provide a valuable mechanism for information exchange [7]. The work presented in this paper is a part of a larger effort to develop methods and tools for a more automated and bidirectional link between product CAD and the different processes and resources applied in APP. This work is based, in its first stage, on extracting assembly knowledge from the CAD file by using feature recognition techniques. The second stage is to model the extracted assembly knowledge to support integration with APP. The third and last stage is to share assembly knowledge via a layered ontology structure, which will serve as a communication mean between assembly design and APP. Figure 1 illustrates the overall integration framework between product design assembly and APP.

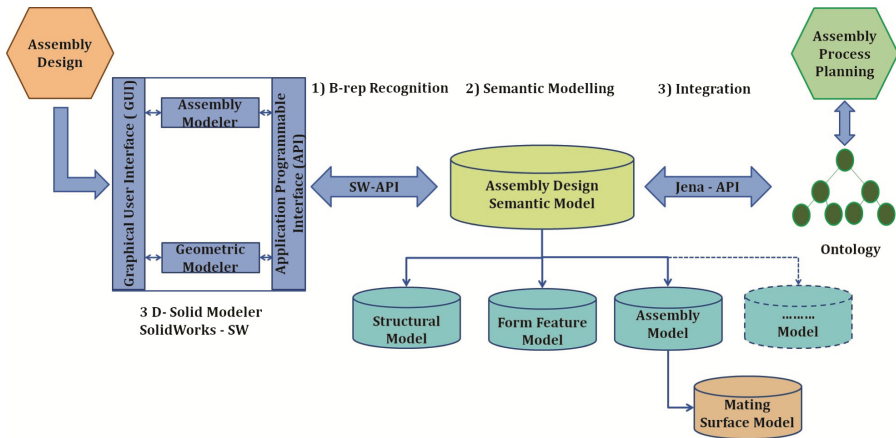


Fig. 1. Proposed framework for integrating product features into APP.

In Fig. 1, the three stages of recognition, modelling, and integration are illustrated. In the first stage, geometrical and assembly knowledge are recognized from SolidWorks (SW) - CAD models. An intermediate modelling stage follows where the recognized knowledge is modelled in assembly design semantic model. Assembly design semantic model is composed from several sub-models to separate the recognized geometry knowledge from assembly knowledge.

Assembly knowledge and geometrical knowledge are modelled in a form feature model and a mating surface model, respectively, while part-assembly relations are modelled in structural model. The last stage will be to structure and store the recognized assembly and geometric semantic knowledge according to a well-defined ontology in order to facilitate sharing and integrating this knowledge with the APP. More details about the proposed integration framework are presented in our previous publications [8–10]. This paper is mainly concerned about the first stage of the integration framework; the recognition of product features from SW - CAD software by using Application Programmable Interface (API).

Product features are classified into low-level features (form features), which are basic geometrical and topological entities such as holes, slots, notches etc. and high-level features, which are characterized by both a form and a specific application

(machining, assembly, tolerance etc.). For example “machining features” can be defined as geometrical and topological entities significant for manufacturing function. The conversion of form features (low-level) into application-specific (high-level) features in terms of functionality, manufacturing and assembly is the overall aim of feature-based modelling [11].

In this context, an assembly feature could be realized as an association between two form features from different parts [12]. Assembly features have been used either in assembly design or assembly process planning. Many definitions are reported in published literature for assembly features either from assembly design or APP perspectives. From assembly design perspective assembly features are “*Mating pairs of form features with parameters and compatibility constraints as part of each feature definition*” [12]. Mating features or conditions are defined as “*relationships that involve contact between parts, as well as relationships in which two parts do not have contact (e.g. clearance conditions!)*” [13].

From an assembly planning perspective, assembly features have been defined as: “*features with significance for assembly processes*” [4]. The same author introduced more specific assembly features, from a process perspective: Connection features “*such as final position, insertion path/point, tolerances*” and handling features “*characteristics that give the locations on an assembly component that can be safely handled by a gripper during assembly!*”. Also more specialized assembly features have been introduced such as joining features. A joining feature has been proposed [14] to represent assembly/joining relations, and it includes joining entities, joining methods, constraints and groove shapes. From a geometrical point of view, feature representation is classified into two types: surface-based or volume-based. Surface features are based on topological entities such as face, edges and vertices with functional meanings on the part boundary; this representation is known as boundary representation (B-rep). Volumetric features are based on three-dimensional geometrical primitives such as sphere and cylinder. This representation is known as constructive solid geometry (CSG). Based on this classification, feature recognition (FR) approaches can be classified as well into B-rep based approaches and CSG. From an engineering point of view FR systems are divided into two methods; external and internal methods [15].

In internal methods the API of the CAD software is used in order to extract topological, geometrical and assembly information relating to a part or an assembly. While in external methods, a CAD model file is exported in a neutral data format (e.g. STEP, IGES, ACIS). The exported file is then translated using compilers (interface programs) to be compatible for a specific application (e.g. commercial CAM system). Both methods have been used by researchers for FR. In our approach to extract product features in mechanical assembly an internal B-rep CAD recognition approach is proposed. The overall approach is illustrated in Fig. 2. In the Fig. 2, four-stage approach of recognition is proposed. In the first stage feature-based modelling; form features from each part in the assembly are recognized, where each part is composed at least from single form feature. The B-rep modelling is the next stage, in which each form features is decomposed into its basic geometrical and topological entities (faces, surfaces, edges, etc.).

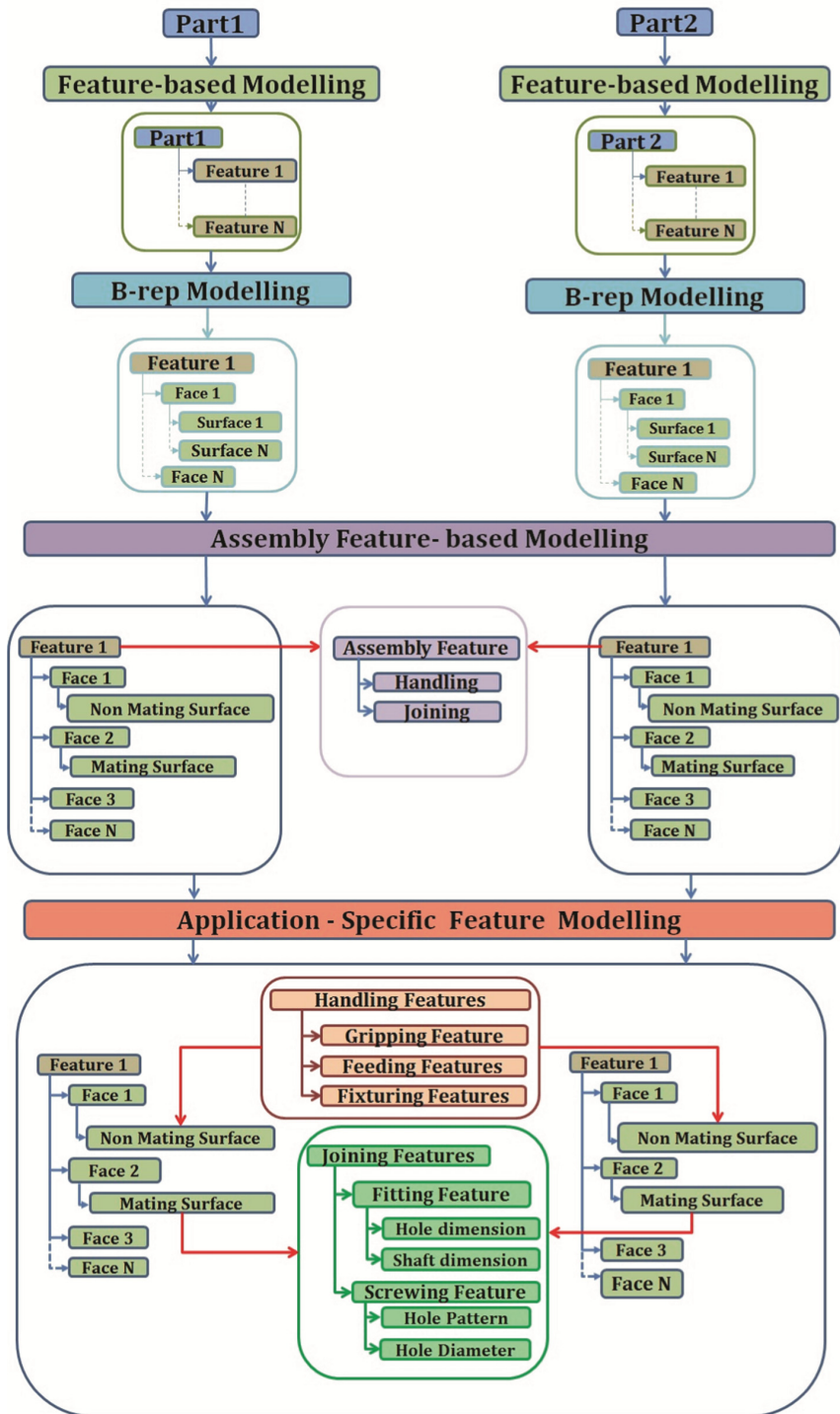


Fig. 2. Proposed methodology for extracting product features in an assembly.

Assembly features modelling is the following stage, in which assembly features (handling for pick and place parts and joining for joining those parts together) are specified based on mating faces or surfaces (surfaces that are involved in joining processes) and non-mating faces or surfaces (surfaces that are not involved in joining process). Further investigation about the geometry and non-geometry (tolerance and positional parameters) for the candidate surfaces included in the assembly processes will take place in the next and final modelling and recognition stage application - specific feature modelling. Based on these details more specific assembly features are determined: gripping, feeding and fixturing will be derived from handling features, and different joining features as screwing, fitting, etc. are derived from joining features, which will be known as fitting features, screwing features and so on. Those features will give a valuable aid in determining the required handling and joining processes needed to assemble a product. The derivation and the extraction of those application-specific features will be our final ultimate aim in this paper by using SW-API programming techniques. The organization of this paper is as follows: Sect. 2 illustrates the relation between the research and the smart systems. Section 3 presents the programming details of the proposed algorithms for features recognition with a case-study example. Section 4 draws a conclusion.

2 Relating Research to Smart Systems

A smart manufacturing environment is an automated or semi-automated system, in which information technology is extensively utilized in design, planning and management. The automation of CAD design by using programming algorithms and the facilitating of knowledge transfer inside the manufacturing environment by using ontologies as in the proposed framework shown in Fig. 1 will provide a valuable aid in creating automated and smart manufacturing environment.

3 SW-API Recognition Approach

In this paper, SW-CAD software is used to construct the automatic feature recognition system and to construct the case-study example. SW-API is an interface that allows exchange of data between CAD design and different software applications. SW-API consists of function calls which are used to access the data structures and their contents from SolidWorks software. The SW-API supports several programming languages such as VBA (Visual Basic for Application). The API is used by writing function calls, which provide linkage to the required subroutine for execution. The topology and geometric data of the CAD assembly in SW is accessed by the proposed algorithm through the API function calls. Microsoft VBA is embedded inside SW CAD software, which enables the recognition process by calling SW-functions from the code written in VBA. Everything in SW is considered an object to the SW-API, those objects are not actually the thing itself, but “interfaces” to those objects. There are three main SW document types: Parts, Drawings and Assemblies. Each document type has its own object (*PartDoc*, *DrawingDoc* and *AssemblyDoc*) with its own set of related objects and functions.

For example the *AssemblyDoc::Mate2* object exists in *AssemblyDoc* document because adding and extracting mate relation is specific to *AssemblyDoc*. Beside the specific objects, which are belonging to a specific document, general objects are available on *ModelDoc2* such as *Feature*, *FeatureManager*, *Configuration*, *SelectManager* and *Sketch*. Those general objects could be accessed by different documents in the SW-API. More details about fundamentals of SW-API are presented in [16]. The case-study model is a three-button panel assembly; the exploded view of the case study assembly is shown in Fig. 3 with its seven parts (the base part, the three buttons, the cover and the screws) and the candidate surfaces, which are significant for the assembly processes. In Fig. 3 for each part some of its surfaces are indicated by labeling the surface with its type (Pla is referred to planner surface and Cyl is referred to circular surface), its part and its number. To make the case-study easier to follow, only the surfaces correspond to one button (red button) and one screw are indicated in the Figure below. For each part in the case-study assembly, the candidate surfaces will be classified into mating surfaces (surfaces involved in mating relations) and handling surfaces (surfaces that will be used in gripping process). Figure 4 illustrates a surface model for the case-study assembly with all the required knowledge needs to be extracted from CAD model in order to facilitate APP.

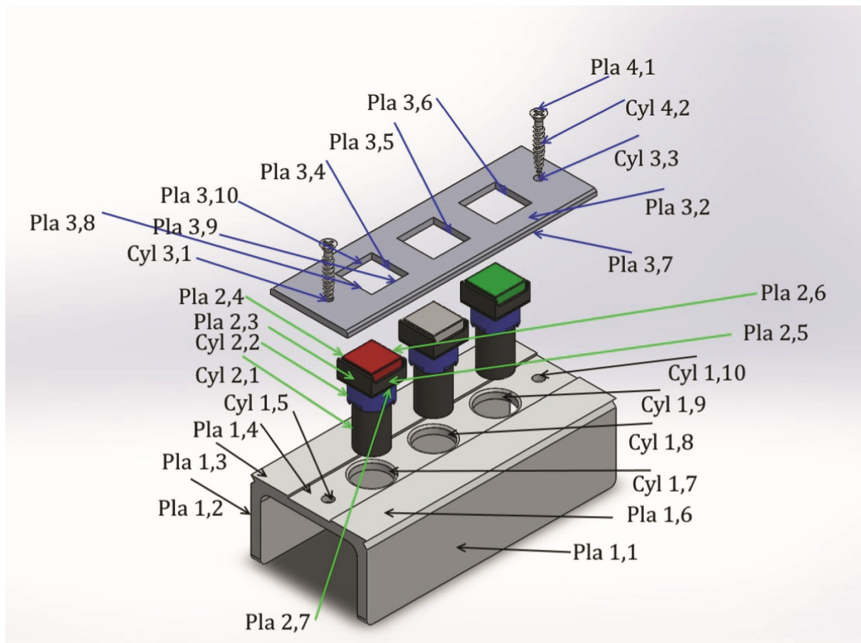


Fig. 3. Exploded view of the three-button assembly with the indicated surfaces

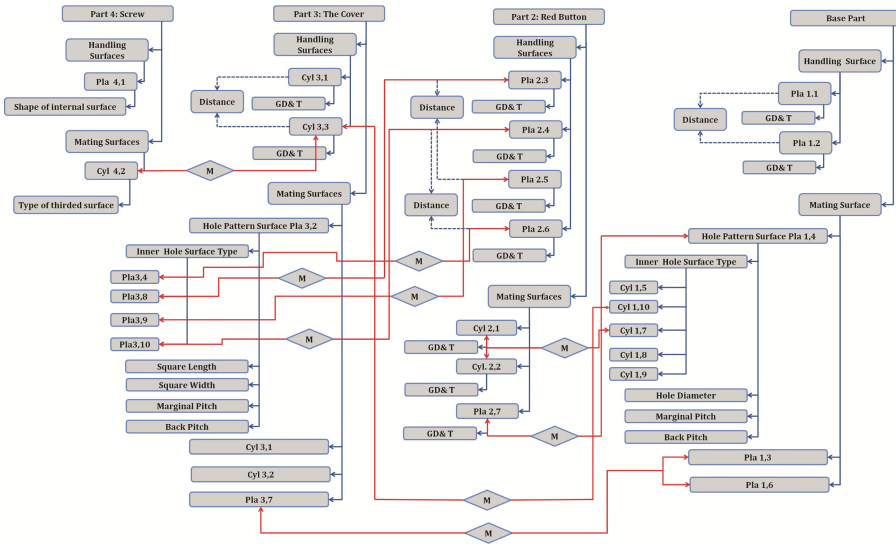


Fig. 4. Surface model for the case-study assembly.

In Fig. 4 all the candidate surfaces for each part of the case-study model is classified into Handling and mating surfaces. Handling surfaces will be chosen according to a criterion mainly depends on geometrical dimension and tolerance (GD&T) information and the distance between the corresponding handling surfaces in order to determine the gripping range of the gripper. The mating surfaces are connected with each other via red lines with M symbol indicates “mates” between the surfaces. The GD&T information will be extracted as well for the mating surfaces to give information about the fitting joining processes that will take place between corresponding mating surfaces. The (GD&T) information will aid in determining fit relations between mating surfaces (surfaces Cyl 2,2 (shaft) and Cyl 1,7 (hole)). Three types of fit relations are mentioned in literature, clearance fit between hole and shaft, which is identified if the minimum allowable dimension of a hole is larger than the maximum allowable dimension of a shaft.

Transition fit, which is identified if the minimum allowable dimension of a hole is smaller than the maximum allowable dimension of a shaft, and the maximum allowable dimension of a hole is larger than the minimum allowable dimension of a shaft. The last type of fit is interference fit, which is identified if the maximum allowable dimension of a hole is smaller than the minimum allowable dimension of a shaft. Identifying fit relations will aid in determining the type of fit process whether its press fitting (for interference fit) or shrink fitting (for other fit types). Another important information that is indicated in Fig. 3 and could be identified from mating surfaces is the hole pattern. A hole pattern has several parameters such as hole diameter, hole type, back pitch (the distance between the centers of successive holes) and marginal pitch (the distance between hole-center and surface’s edge) that could help to determine a specific joining

process. One of these attributes is the hole type. Identification of a hole as threaded determines screwing as a joining process to be selected in APP.

A high-level flowchart of the recognition algorithm is shown in Fig. 5. This high level description consists of nine steps. The first step is to read the CAD assembly file from SolidWorks software. The second step is to extract all the surfaces from the assembly by traversing the feature manager tree using the SW-API functions and methods shown beside the corresponding block in the Fig. 5. The third step is to extract the corresponding dimensions and tolerances for all the surfaces in the assembly. Each of the methods and objects indicated beside the related block has access to a specific type of dimension or geometrical tolerance. Dimensions and tolerances have to be assigned to the SW-CAD part document by using DimXpert manager in order to be extracted by this method. The fourth step is to get Persistent IDs for all the surfaces extracted in the second step.

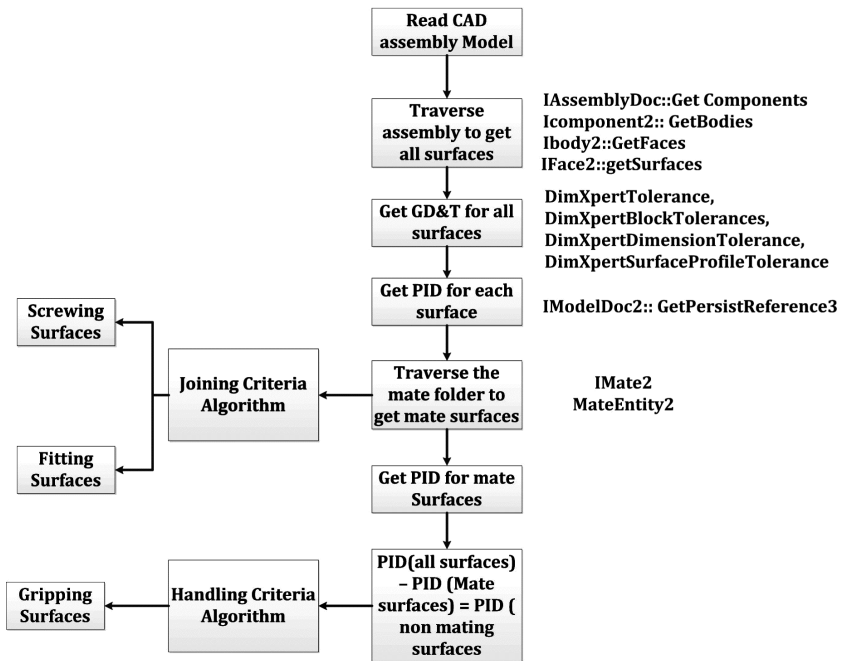


Fig. 5. High level description of the recognition algorithm.

Persistent IDs (PID) are a unique identifier for any object and they will be used to identify surfaces in the assembly. The fifth step is to extract the mating surfaces by locating the mate folder in the feature manager tree using feature traversal. This can be achieved by using *IFeature::GetSpecificFeature2* to get *IMate2*, then getting information about the mates, like the type (*IMate2::Type*) and the corresponding reference mating entities. This can be done by using *IMate2::MateEntity*, which will give a pointer to *IMateEntity2*, then use *IMateEntity2::Reference* to get the pointer

to the actual reference entity. The sixth step is to get PID for each mate surface extracted in the previous step.

The seventh step is to find the non-mating surfaces by subtract the PID of all surfaces from the PID of the mate surfaces, the result will be the PID for the non-mating surfaces. The eighth step will be to investigate the non-mating surfaces to determine the handling surfaces; this will be achieved by determining the handling criteria. The handling criteria will be to find the non-adjacent parallel surfaces, which are suitable for gripping. Additional parameters in these criteria will be a threshold for the surface profile tolerance for those surfaces. Surfaces with surface profile within a specified range will be chosen for gripping and will be indicated as gripping surfaces.

The ninth and the last step will be the joining criteria for the mating surfaces. The joining criteria will be based mainly on the dimensions of the cylindrical mating surfaces to determine the type of fitting joining processes and on the type of the mating surface if it is threaded or not threaded to distinguish between fitting processes and screwing processes. The final ultimate aim of these criteria is to determine the screwing surfaces that will be used in screwing assembly processes and fitting surfaces that will be used in different fitting assembly processes.

4 Conclusion

In this paper, an attempt to create internal-B-rep recognition system to support APP is proposed. Future work includes updating the algorithm in Handling and joining criteria to support more handling and joining processes.

References

1. Cho, S.: Mechanical Assembly. The Mechanical Engineering Handbook Series, CRC Press LLC, Boca Raton (2005)
2. Wang, L., Keshavarzmanesh, S., Feng, H.-Y., Buchal, R.O.: Assembly process planning and its future in collaborative manufacturing: a review. *Int. J. Adv. Manuf. Technol.* **41**(1–2), 132–144 (2008)
3. Du, H., Zha, X.F.: A PDES/STEP-based model and system for concurrent integrated design and assembly planning. *Comput. Aided Des.* **34**, 1087–1110 (2002)
4. Van Holland, W.: Assembly features in modelling and planning. Ph.D. thesis, Delft University of Technology, Delft (1997)
5. Bronsvort, W., Van Holland, W., Jansen, F.: Feature modelling for assembly. In: Straßer, W., Wahl, F. (eds.) *Graphics and Robotics*, pp. 131–148. Springer, Heidelberg (1995)
6. Michael, L.: *Planning and Scheduling in Manufacturing and Services*. Springer, Heidelberg (2009)
7. Dartigues, C., Ghodous, P., Gruninger, M., Pallez, D., Sriram, R.: CAD/CAPP integration using feature ontology. *Concurr. Eng.* **15**(2), 237–249 (2007)
8. Hasan, B., Wikander, J.: Product feature modelling for integrating product design and assembly process planning. *World Acad. Sci. Eng. Technol. Int. Sci. Index* 118, *Int. J. Mech. Aerosp. Ind. Mechatron. Manuf. Eng.* **10**(10), 1725–1735 (2016)
9. Hasan, B., Wikander, J., Onori, M.: Assembly design semantic recognition using SolidWorks-API. *Int. J. Mech. Eng. Robot. Res.* **5**(4), 280–287 (2016)

10. Hasan, B., Wikander, J., Onori, M.: Ontological approach to share product design semantics for an assembly. In: *Proceedings of 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, vol. 2, pp. 104–111 (2016)
11. Stefano, P., Bianconi, F., Angelo, L.: An approach for feature semantics recognition in geometric models. *Comput.-Aided Des.* **36**, 993–1009 (2004)
12. Shah, J., Rogers, M.: Assembly modeling as an extension of feature-based design. *Res. Eng. Des.* **5**, 218–237 (1993)
13. Mullins, S., Anderson, D.: Automatic identification of geometric constraints in mechanical assemblies. *Comput.-Aided Des.* **30**(9), 715–726 (1998)
14. Kim, K.: Assembly operation tools for e product design and realization. Ph.D. dissertation, University of Pittsburgh, Pittsburgh (2003)
15. Miao, H.K., Sridharan, N., Shah, J.J.: CAD-CAM integration using machining features. *Int. J. Comput. Integr. Manuf.* **15**, 296–318 (2002)
16. SolidWorks Help (2017). <http://help.solidworks.com>