EXTRACTING ENGINEERING FEATURES FROM B-REP GEOMETRIC MODELS

Christian Van der Velden *, Hao-Lan Zhang *, Xinghuo Yu *, Tim Jones **, Ian Fieldhouse **, Cees Bil ***
*** Electrical and Computer Engineering, RMIT University, Melbourne, Australia,**
**** GKN Aerospace Engineering Services Pty. Ltd., Port Melbourne, Australia**
***** School of Aerospace, Mechanical and Manufacturing Engineering, RMIT University, Melbourne, Australia**

Keywords: *Automated Feature Recognition (AFR), Computer Aided Design (CAD), Knowledge Based Systems*

Abstract

Inefficient integration between engineering software tools, particularly Computer Aided Design / Analysis / Manufacturing (CAD / CAE / CAM) systems, continues to be a significant problem for many engineering sectors. Transfer of model data between tools is often through neutral Boundary Representation (B-Rep) formats such as STEP which transfer model geometry only. B-Rep formats do not preserve modelling information and design intent added by high level tools (such as design-by-feature and parametric information captured by CAD systems), and omit information about significant model features from other engineering contexts (e.g. analysis and manufacturing). Automatic Feature Recognition (AFR) technologies have been suggested as a solution to engineering tool integration issues and while they have been shown to be successful in integrating CAD / CAM systems, they have limited application to CAE tools. This paper introduces a new AFR framework for extracting analysis features from B-Rep models (in STEP format). The framework is implemented as a prototype software platform that demonstrates extraction of analysis features from integrally stiffened frames.

1 Introduction

The detailed development phase of modern engineering design, analysis and manufacturing processes is largely driven by software including CAD, CAE, and CAM systems. The exchange of data between these systems is often through neutral B-Rep formats such as STEP. Such formats preserve only model geometry, omitting information relating to modelling processes, design intent and significant features from different engineering contexts (design features, analysis features and manufacturing features).

A common consequence of inefficient data transfer between tools is time consuming manual manipulation of model geometry and data sets to construct a model for the downstream engineering process. For example, a Finite Element Analysis (FEA) model often contains approximations of analysis features to improve analysis efficiency (e.g. a thin section of material can be modelled as a set 2D elements with equivalent thickness rather than 3D elements). This approximation processes is highly manual and time consuming with analysts interacting with only raw geometry (e.g. extraction of mid-surfaces, manual measurements within the model, etc.).

Automated Feature Recognition (AFR) technologies have been suggested as a possible solution to tool integration inefficiencies, and have been shown to be successful in CAM applications [1-2]. Automatic extraction of features relevant to the downstream process reduces the amount of manual work required to construct models in different software platforms. However, the majority of existing approaches are targeted at CAM systems and have limited applicability to the CAE example discussed above.

This research aims to develop a framework for automatically extracting engineering features from neutral B-Rep models for use in downstream processes including, but not limited to, analysis (CAE systems) and manufacturing process planning (CAM systems). This paper introduces the AFR framework and its implementation in a prototype system which extracts analysis features from integrally stiffened frames. The framework and software platform have potential to significantly reduce low level modelling tasks, providing time and cost savings.

This paper is organized as follows: Section 2 provides a brief overview of the problem including the concept of engineering features and common techniques for identifying features in geometric models. Section 3 introduces the AFR framework including the feature recognition process, feature extraction mechanism and methodology for structuring feature rules. Section 4 presents results of the prototype AFR system in relation to simple and complex industrial test cases before concluding in section 5.

2 Problem Background

2.1 Engineering Features

Engineers interpret a component as a collection of features that relate to its function or fabrication. The definition of these features may vary depending on the engineering discipline (e.g. design, analysis, manufacturing, etc.). Engineers often have little need to consider how the part is represented within software.

Consider the following example. A detailed design model of an integrally stiffened frame (Fig. 1) was modelled in CAD software and represents the desired geometry of the completed component.

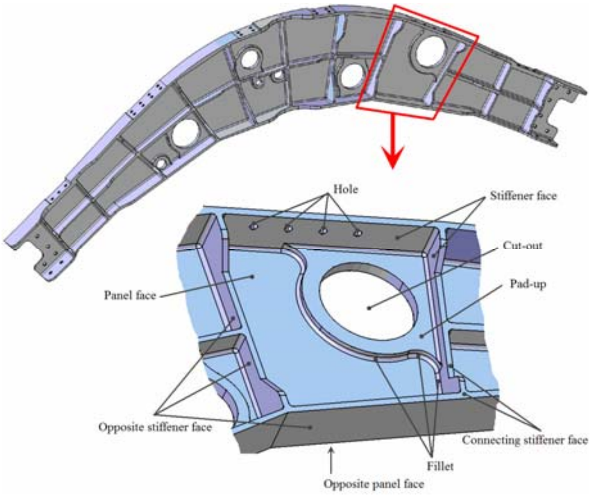

Fig. 1: Integrally stiffened frame and panel feature

An analysis model of the component must be constructed to verify its performance against expected load spectrum. The analyst considers the component to be a collection of analysis features including panels and ribs, which perform different load carrying functions (a panel and surrounding ribs is enlarged in Fig. 1). The analysis model approximates detailed design geometry using idealised analysis features which are analysed using company methods. Analysis features are typically constructed by extracting parameters from detailed model geometry (e.g. physical dimensions, extraction of mid-surfaces, etc.). This parameter extraction is typically highly manual, and often involves the analyst moving back and forth between CAD and CAE tools.

Although the use of neutral B-Rep format ensures geometry can be transferred between most CAD / CAE / CAM systems, the loss of modelling data and omission of significant features in different contexts results in a large amount of low-level manual work. There is a clear case for better integration of engineering tools, particularly the transfer of model data. Because there are a large number of commercial CAD / CAM / CAE packages competing within the market place, improvements in one system would not necessarily be immediately available to engineers due to economic factors and commercial arrangements with software vendors. Automatic recognition of engineering features from neutral B-Rep models can provide a practical solution to reducing manual model manipulation resulting from inefficient data transfer between tools that can be tailored to the needs of the downstream engineering process.

2.1 Automated Feature Recognition Techniques

AFR technologies have been suggested as a possible solution to engineering tool integration inefficiencies. The extraction of features reduces the need to manually modify input data for different tools, and definition and representation of features can be tailored to the relevant discipline or application.

Two main techniques characterize the field of feature recognition: (1) Volumetric Decomposition and (2) Adjacency-Based methods, both of which use rules to identify instances of predefined features in model data.

2.2.1 Volumetric Decomposition

Volumetric decomposition methods attempt to reduce a component to a collection of elementary volumes that are added to or subtracted from a starting solid [3,4]. Analysis of the arrangement of elementary volumes produces a representation of features. Consider the simple example in Fig. 2. A simple structure consisting of two "pocket" features is shown in the upper image. A pocket feature can be described as a volume of material removed from a solid forming a depression. It can be represented by either the volume that is subtracted from a solid to result in the feature, or by the particular pattern of faces surrounding the feature itself. The centre image shows the initial volume and the lower image show two volumes that are subtracted from this to result in the final structure.

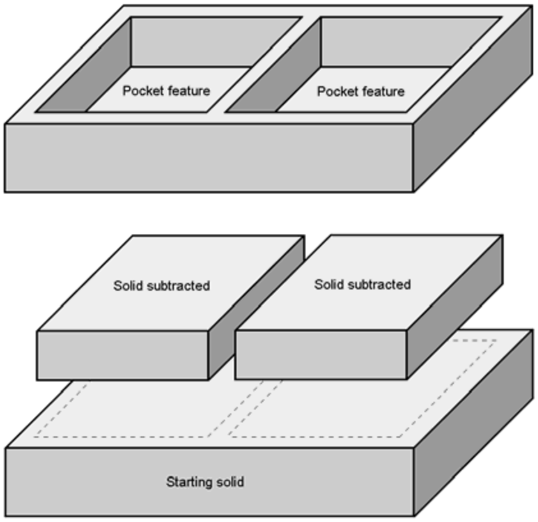

Fig. 2 Volumetric decomposition of a simple frame

The two main difficulties with volumetric decomposition methods include:

- Volumetric methods rely on the ability to decompose a geometric model into primitive solids (i.e. polyhedral shapes). Models in many engineering disciplines contain complex curve and surface geometry making it difficult or impossible to identify these basic shapes.
- The order of decomposition of a component can influence the collection of primitive shapes that are identified, making it difficult to control the pattern matching process.

2.2.2 Adjacency-Based Methods

Adjacency methods search for feature patterns in a graph (or matrix) based representation of the B-Rep model, and are the most common approach to feature recognition as graph theory is well understood. This approach uses an Adjacency Attribute Graph (AAG) in which nodes represent faces of a B-Rep model and graph edges represent adjacency relationships (shared edges) between connected face pairs [3-8]. Graph edges are labelled with attributes describing the connection between two faces (usually concave (0) or convex(1)).

Patterns corresponding to features are identified prior to feature recognition [9] and the graph is searched for instances of these patterns. For example, a pocket is defined by *"the union* *of a central face and its adjacent faces where all adjacency attributes are concave*". The structure used in the above example is shown in Fig. 3 with corresponding AAG with pocket features indicated in red.

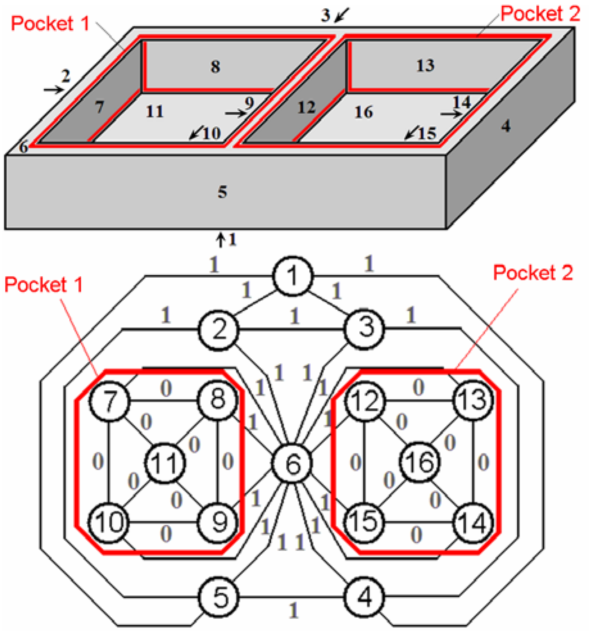

Fig. 3 AAG of simple frame with pocket features indicated in red

This basic technique has a number of drawbacks that have limited its success in commercial systems:

- The inclusion and interaction of detailed features complicate the graph structure such that predefined feature patterns are not able to describe features. If a component has a different surface combination to that stored in the rule base (e.g. a new face introduced when an edge fillet is created), the feature will not be identified.
- Systems are often limited to relatively simple polyhedral or 2.5 dimensional (finite layer) objects due to the difficulty in determining adjacency attributes (graph edges labels) for complex surfaces such as B-Spline based surfaces.

3. An Automated Feature Recognition Framework

3.1. Application of the Framework

AFR technology has many possible applications in several engineering contexts including automatic model construction (e.g. design, analysis, NC machining), model checking (against design rules or other criteria), manufacturing process planning, and others. The AFR framework developed through this research supports numerous applications of the same principles.

A prototype system implementing the framework was developed to demonstrate feature recognition principles. This prototype system was based on a particular application: identification of analysis features in integrally stiffened frames. The system is expected to automate extraction of analysis feature data including panels and ribs to facilitate automation in downstream stress analysis processes.

A typical component that will be interrogated by the AFR system is shown above in Fig. 1. This component is a detailed design model of a integrally stiffened frame, consisting of a several panel and rib features with numerous detailed features (e.g. fillets, chamfers, holes, pad-ups). The stress analysis process for structures of this type requires a model to be constructed from the analysis features comprising the structure. The main difficulty lies in the fact that analysis features are defined from a different engineering context than that used to produce the detailed design model, and direct extraction of the feature is usually not possible. The result is that analysis features have to be manually specified using the detailed design geometry as a reference. The aim of the prototype system is to extract analysis feature geometry to facilitate automatic construction of the analysis model.

3.2. Framework Architecture

The extraction process under the AFR framework developed through this research is summarised in Fig. 4. Firstly, a detailed Feature Recognition (FR) model is constructed from topologic and geometric data extracted from the input CAD file which is processed to determine detailed quantities between entities related by adjacency and hierarchical relationships. These quantities include angular tests, tangency tests, surface area tests and others. The extension of adjacency relationships to include entity pairs within a local vicinity (which may be separated by one or more intermediate objects) significantly improves the parameters available to formulate feature rules.

Rules describing attributes of features are formulated and implemented in an inference engine which searches the FR model dataset to identify instance of features. Feature rules are generic, including parameters to identify feature attributes independent of local detail (fillets, holes, etc.).

The representation of extracted features is tailored to the particular downstream tool/process (e.g. full feature geometry, feature mid-surface, dimensions and location, and others).

The AFR framework is divided into 5 functional models: (1) FR model representation (2) System framework, (3) Geometry Engine, (4) Inference Engine and (5) Feature Representation. The design of this framework is independent of the implementation of each module.

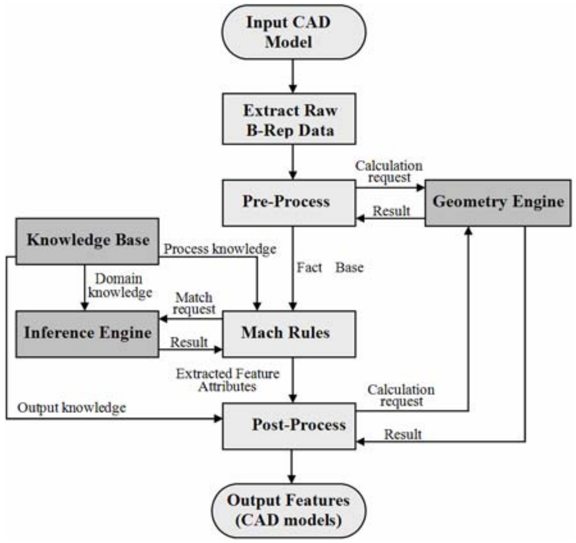

Fig. 4: Automated feature recognition process

3.3 Key Feature Extraction Mechanisms

The feature recognition technique developed for this system differs from traditional graph techniques in several key areas:

- An analysis of relationships between faces that are not immediately adjacent provides a more complete view of the model allowing the interaction of detailed features to be determined with a greater level of accuracy.
- Identification and suppression of detailed features such as fillets, rounds, holes and thickness changes simplifies the model. This allows generic rules to be specified that can identify features regardless of the complexity of surrounding detail.
- The system employs a computational geometry engine to determine relationships between faces and edges with complex underlying surface and curve geometry. This allows complex 3D design models to be evaluated.
- High computational requirements associated with pattern matching in graph structures are reduced with the integration of an inference engine that efficiently searches model data for entities that satisfy feature rules.
- The use of a neutral geometry format reduces the dependency of the feature evaluation on modelling processes and third party engineering tools.

3.4 Developing and Structuring Feature Recognition Rules

The identification, capture, organisation and implementation of feature recognition rules within the AFR system is an important design consideration that affects system performance and accuracy of feature searches. This section describes a five step process for structuring feature recognition rules covering these aspects (Fig.3). The rule structuring techniques form part of the AFR framework and were implemented in developing rules to identify panel and rib analysis features.

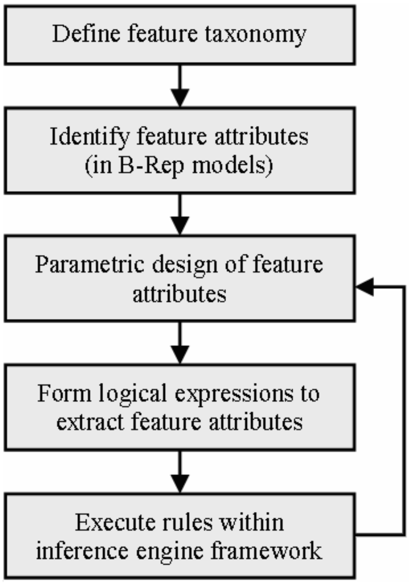

Fig. 5: Rule design process

3.4.1 Define Feature Taxonomy

Feature taxonomy refers to the fundamental characteristics of a given feature type – i.e. how is a region of a geometric model recognisable as an instance of a feature type? Definition of feature taxonomy is the first step in modelling a feature rule. The definition should be generic, including only the minimum characteristics required to identify an instance of the feature type – i.e. ignore variations caused by sub features.

The application of the AFR framework considered in this paper is extraction of panel and rib analysis features from integrally stiffened frames. This paper will focus on panel extraction. Panels carry a particular mode of load and have specific stress analysis methods. The top level description of a panel is a "*thin planar section of material bounded on all sides by ribs*". This definition is true for all panels regardless of the level of complexity. A simple example is given in Fig. 6. The region highlighted in dark grey is a single panel.

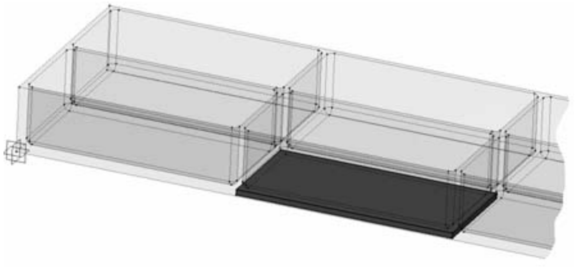

Fig. 6 Simple frame (panel feature in dark grey)

The desired representation of a feature is dependant upon the requirements of the process that will use AFR outputs. In this case the panel is considered to extend from the centreline/surface of surrounding ribs.

3.4.2 Identify Feature Attributes

The second task identifies the minimum set of information that is required to represent the feature that can be identified in a B-Rep model.

Because the AFR system applies inference procedures to the set of processed model data, consequently only this data can be used to define feature attributes. This means that rules can only use topological and geometric entities contained within the model itself, and not extrapolation/interpolation of these entities (e.g. mid-surfaces, intersection curves/points, etc.) unless previously calculated. For this reason, this second task identifies model elements (surfaces, curves, etc.) that provide the necessary information to extract the desired feature. This process involves analysis of a representative set of features covering the range of complexity that can be expected to be processed by the system. This will usually involve the analysis of a number of industrial test cases.

Two panel features from Fig. 1 are shown enlarged in Fig. 7 and Fig. 8. Fig. 7 shows a relatively simple panel, surrounded by four planar stiffeners with edge and corner fillets connecting the planar surfaces. Fig. 8 shows a more complex example where a number of detailed features interact within the panel region.

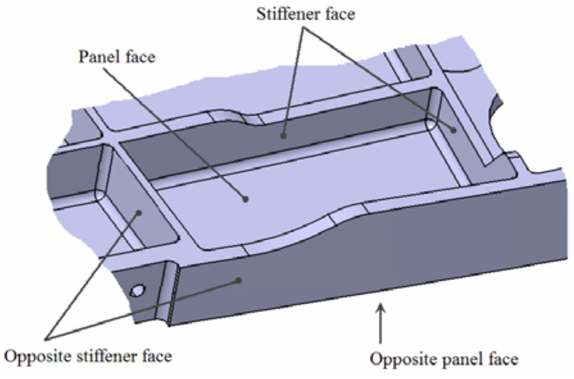

Fig. 7 Key attributes of a simple flat panel

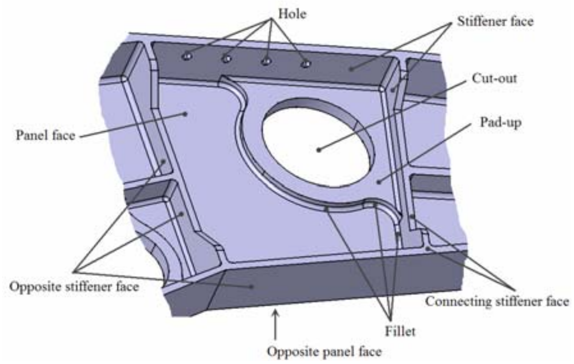

Fig. 8 Key attributes of a complex panel

Considering a number of examples from numerous test cases and the desired panel representation described above, the following attributes represent the minimum information specify panel features:

1) Panel face
2) Opposing panel face
3) Collection of rib faces
4) Collection of opposing rib faces

If, for example, the panel boundary was to be defined from the inside face of surrounding ribs, only the first three attributes would be required to identify the panel.

The panel and opposing panel faces determine the upper and lower panel limits, from which thickness can be determined. Stiffener and corresponding opposing rib faces are used to define the rib centreline which forms the panel boundary. Despite the higher level of complexity of the panel in Fig. 8 than Fig. 7, the same fundamental feature characteristics are preserved.

3.4.3 Parametric Design of Feature Attributes

The parametric design phase identifies parameters of feature attributes that uniquely separate them from other entities in the model data. When identified, attribute parameters are related information contained within the feature recognition model dataset which includes the following parameters:

- Adjacent faces (either faces immediately adjacent, or faces separated by one or more intermediate faces).
- Face surface type
- Edge attribute (concave/convex/tangent)
- Angle between faces
- Relationships with detailed features
- Distance between entities
- Face area
- Face aspect ratio
- Edge curve type
- Edge length
- Surface/curve normal vectors
- Number of wires (edge loops)
- Number of edges in a wire
- Others…

When unique feature attribute parameters are identified, a rule for extracting the feature attribute is written. Each rule consists of one or more constraint statements that use parameters listed above. The general structure for a single constraint statement consist of the following:

- **Search entity:** face, edge, surface, curve, etc. (to be returned)
- **Property type:** surfaceType, curveType, nbWires, angle, edgeAttribute, etc.
- **Operator:** =, >, >=, <, <=, <>, etc.
- **Value:** number, Boolean, surface/curve type, etc.
- **Related Entity:** specific entity, all adjacent entities, parent entities, child entities, etc. (for relationship parameters)
- **Additional condition:** all, any, none, etc.

Typical components that will be interrogated by the AFR system will have a high level of complexity. Consequently the identification of parameters that uniquely describe feature attributes is a manual process. The parametric design process involves analysing a representative set of features and identifying common elements that distinguish them from other model components.

Continuing with the example introduced earlier, instances of Panel Face attributes can be identified from most others using the following criteria: "*a planar face surrounded on all sides of the outer boundary by faces that are concave and normal (within tolerance) to the face surface*", i.e.:

1) surfaceType = plane
2) edgeAttribute = concave for all adjacentFaces on wire 1*
3) faceAngle = 90 degrees (± tolerance) for all AdjacentFaces on Wire 1*
4) faceArea > minPanelArea
*ignoring the presence of detailed features

This simple rule does not account for the presence of detailed features. The actual Panel Face rule implemented in the AFR system is more detailed, and applies equally to components with or without detailed features.

3.4.4 Form Logical Expressions to Extract Feature Attributes

The fourth task translates parameterised rules to a form that can be interpreted by the inference engine, i.e. construction of an executable knowledge base. This is accomplished by forming a set of logical expressions that implement constraint statements in parameterised rules developed in the previous step.

An implementation of the Prolog rule engine language was selected as the inference engine platform. This task therefore involves writing the logical expressions in a Prolog-readable form.

Continuing with the example, the logical expressions for the rule defined in the previous step are as follows:

```
get list of all faces with surfaceType = plane (List1);
for each face in List1
    if faceArea > minPanelFaceArea then
        get list of all adjacent faces on outer wire (List2);
        get list of all adjacent faces on outer wire that are
         normal and concave (List3);
        if lengths of List2 and List3 are NOT equal then
            remove current face from List1;
        end if
    end if
end for
return List1;
```

3.4.5 Execute Rules within Rule Engine Framework

Outputs from the inference engine will be a list of entities that satisfy criteria contained with in the rule for a particular feature attribute. This will usually be a list of unique identifiers of model entities.

A number of industrial test cases were used to develop the executable knowledge base for extracting analysis features from integrally stiffened frames (such as Fig. 4).

A method of testing rules was developed as part of the AFR system framework. It allows entities to be output as a CAD-readable file that can be viewed alongside the original model to determine if the function has performed as expected.

The rule structuring methodology is expected to be an iterative process. In some cases refinement of parametric design or formation of logical expressions may be required to finalise a feature attribute rule.

4 Results from System Testing

The AFR framework was implemented as a software platform to demonstrate rule based feature recognition. The resulting AFR system accepts a STEP model input and superimposes identified features over the original part geometry. In this case feature geometry is output as individual STEP models, however, the output can be tailored to the needs of the downstream engineering process.

Two test cases are described in this section: a simple test case that demonstrates the generic nature of the rule base, and a complex industrial test component.

4.1 Simple model

A simple rectangular frame with simple underlying surface and curve geometry is shown in Fig. 10 (upper image). This basic frame features a number of panel and rib analysis features. A number of variation of the basic frame geometry were made to test the ability of feature rules to extract analysis features regardless of the level of local detail including fillets, holes, and pad-ups. Fig. 10 (lower image) shows the same frame with local detail and flanges on each end of the frame.

Because the same basic frame geometry is preserved in each variation, expected output for each variant is identical.

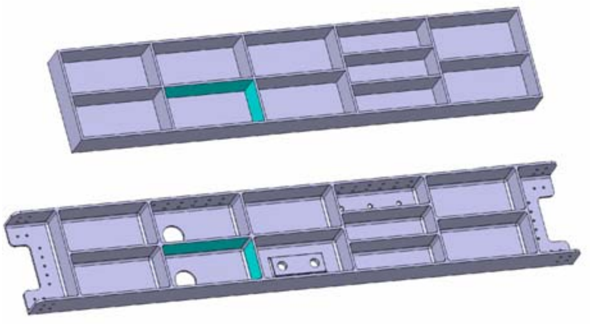

Fig. 9: Automated feature recognition process

The AFR system was tested with each variation of the frame. In all cases, the system successfully identifies all panels and rib instances and produces a representation consistent with the definition given above (i.e. panel boundaries extend to the centreline of surrounding ribs, and rib features extend to the centreline of adjacent ribs).

Fig. 10 shows a screen capture of the AFR system showing extracted features for the test case (variation with detailed features). Identified features are listed in a tree which indicates model entities corresponding to attributes of each feature instance. Users can select and highlight individual features. A selected panel is shown in blue.

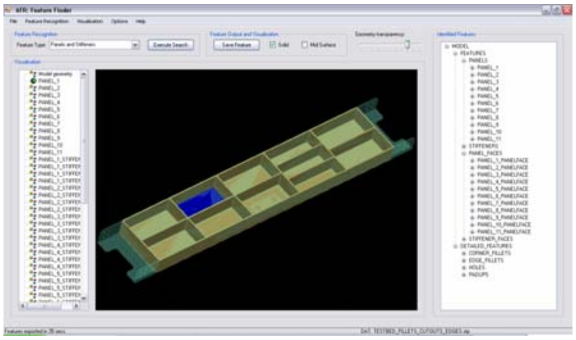

Fig. 10: Automated feature recognition process

4.2 Complex Test Case

The complex model introduced in Fig. 1 was also tested. The AFR system identified 14 out of the total 19 panels. A screen capture of identified features is given in Fig. 11.

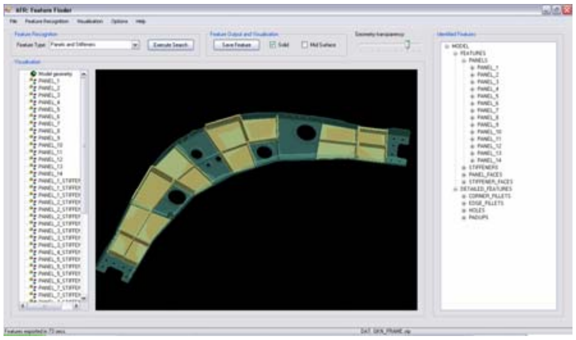

Fig. 11: Automated feature recognition process

In all cases where panels were not identified, pad-up local features (areas of increased thickness) were present. This result was expected as the rule governing Panel Faces does not yet take into account the presence of pad-up features.

To enable a generic set of rules to be applied to models regardless of the level of complexity, sub rules must exist that govern the behaviour the search algorithm when detailed features exist in the local search vicinity. For example when an edge fillet is reached, a sub rule exists to use the other face connected to the edge fillet rather than the fillet face itself. A similar rule exists for pad-up features, but has not yet been implemented. It is expected that when this rule has been implemented, identification of all panel and rib features will be possible.

5 Conclusion

A framework for recognising and extracting engineering features from geometric models has been developed and implemented in a software system that automates the feature recognition task. The research has shown that a rule based approach to feature recognition is feasible.

Feature extraction techniques employed by the AFR system described in this paper provide several advantages over many existing techniques. The integration of a computational geometry engine extends extraction techniques to components with complex surface and curve geometry. An enriched model dataset that includes more information about the model than provided in previous methodologies allows high level feature rules to be developed. The identification and suppression of detailed features simplifies the model and facilitates the specification of generic feature rules that apply to features with varying detail.

The prototype AFR system is currently capable of extracting of panel and rib analysis features from integrally stiffened frames. Results of system testing described in the previous section show full recognition of panel and rib features from frames with simple geometry, and partial recognition from complex models.

Further algorithm development in areas of (1) Geometry Engine and (2) Inference Engine will improve accuracy and robustness of the tool. Collection of new test cases will allow feature attribute rules to be improved, allowing complex panel/rib features to be recognised fully, and rules for new feature types to be specified.

Acknowledgements

This research is a partnership between RMIT University and GKN Aerospace Engineering Services Pty. Ltd., with funding from the Commonwealth of Australia, through the Cooperative Research Centre for Advanced Automotive Technology (AutoCRC).

References

[1] A. D. McCormack, R. N. Ibrahim, "Process planning using adjacency-based feature extraction", *International Journal of Advanced Manufacturing Technology*, Vol. 20, pp. 817–823, Springer-Verlag London, 2002.

[2] M. P. Bhandarkar, R. Nagi, "STEP-based feature extraction from STEP geometry for Agile Manufacturing", *Computers in Industry*, Vol. 41, pp. 3–24, 2000.

[3] J. Han, M. Pratt, W. C. Regli, "Manufacturing feature recognition from solid models: A Status Report", *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 6, 2000.

[4] B. Li, J. Liu, "Detail feature recognition and decomposition in solid model", *Computer Aided Design*, Vol. 34, pp. 405-414, 2002.

[5] P. Holland, P. M. Standring, H. Long, D. J. Mynors, "Feature extraction from STEP (ISO 10303) CAD drawing files for metalforming process selection in an integrated design system", *Journal of Materials Processing Technology*, Vol. 125-126, pp. 446–455, 2002.

[6] S. Venkataraman, M. Sohoni, V. Kulkarni, "A graph-based framework for feature recognition". *Solid Modeling*, ACM, Michigan, 2001.

[7] J. Comey, D. E. R. Clark, "Efficient face-based feature recognition", Herriot-Watt University, 1993.

[8] H. Lockett, D. Guenov, "Graph-based feature recognition for injection moulding based on a mid-surface approach", *Computer Aided Design*, Vol. 37, pp. 251-262, 2005.

[9] E. Brousseau, S. Dimov, R. Setchi, "Knowledge acquisition techniques for feature recognition in CAD models", *Journal of Intelligent Manufacturing*, Vol 19, pp. 21-32, Springer Science+Business Media, 2008.

Contact Author Email Address

Xinghuo Yu x.yu@rmit.edu.au

Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS2010 proceedings or as individual off-prints from the proceedings.