Int J Adv Manuf Technol (2015) 76:2159–2183
DOI 10.1007/s00170-014-6396-0

ORIGINAL ARTICLE

Feature-based design approach for integrated CAD and computer-aided inspection planning

Ali Kamrani · Emad Abouel Nasr · Abdulrahman Al-Ahmari · Osama Abdulhameed · Syed Hammad Mian

Received: 15 November 2013 / Accepted: 15 September 2014 / Published online: 10 October 2014

Abstract Coordinate-measuring machines (CMMs) have widely been utilized for performing inspections due to its high precision and accuracy. With increasing complexities of parts and tighter tolerances, there is a need of well-defined strategies and techniques to effectively plan inspection process for CMMs. The proposed work comprehensively describes different steps to generate an instruction file which can finally be executed on CMMs for efficient measurement. This work which commences with feature extraction and recognition followed by generation of probe approach directions for accessibility, setup planning to get part orientation, determination, and distribution of touch points on features including other activities leads to an effective inspection plan. The inspection plan is then used to produce DMIS file which after post-processing is executed on CMM. A series of techniques and algorithms such as clustering algorithm, graphical method, artificial neural networks, and algorithms for generation of touch points have been utilized to accomplish efficient inspection on CMM. The effectiveness of proposed strategies and techniques has also been verified through a part case study.

Keywords Coordinate-measuring machine (CMM) · Computer-aided inspection planning (CAIP) · Setup planning · Probe approach direction (PAD) · Touch points · Dimensional measuring interface standard (DMIS)

A. Kamrani (✉)
University of Houston, Houston, TX, USA
e-mail: akamrani@uh.edu

E. Abouel Nasr · A. Al-Ahmari · O. Abdulhameed · S. H. Mian
Industrial Engineering Department, College of Engineering, King Saud University, Saudi Arabia, Riyadh, Saudi Arabia

E. Abouel Nasr
e-mail: eabdelghany@ksu.edu.sa

A. Al-Ahmari
e-mail: alahmari@ksu.edu.sa

O. Abdulhameed
e-mail: osamaie@gmail.com

S. H. Mian
e-mail: smien@ksu.edu.sa

E. Abouel Nasr
Faculty of Engineering, Mechanical Engineering Department, Helwan University, Cairo, Egypt

1 Introduction

Increased demands for accuracy and precision make inspection an indispensable step in modern manufacturing industries. In fact, high-quality requirements by customers have established inspection as one of the most vital processes in production industries. Inspection compares manufactured products with its standard drawing to verify whether they lie within specified tolerance or not. With increased emphasis on accuracy and precision, computerization of inspection resulting in computer-aided inspection (CAI) has been of utmost importance. The coordinate measurement machine (CMM) has been one of the most sought after inspection tools for CAI due to its high accuracy. Inspection through CMM requires a large amount of design information for the part which it collects from the CAD systems through a link known as computer-aided inspection planning (CAIP). Information gathered from the CAD systems is transferred to CMM through CAIP for part inspection. It has been recognized that CAIP is very critical for the proper integration of CAD and inspection systems such as CMM. A well-developed CAIP enables efficient exchange of information between planning, design, and inspection activities that increase inspection efficiency and effectiveness. It is of utmost importance to develop an effective inspection planning strategy, i.e., CAIP for the measurement of complex parts on CMMs. Effective CAIP ensures reduced inspection time, improved consistency, and reliability of measurement results. CAIP consists of all information and decisions regarding probing features and orientation, setup planning, probe accessibility, probe selection and orientation, collision avoidance, identified coordinates and number of points for part features, probe execution, etc. According to Limaiem et al. [1], generation of any inspection plan consists of four fundamental activities including accessibility analysis of the part and its measurement points, clustering of measurement points, sequencing of measurement points, and a collision-free probe path generation.

Inspection on CMM is a complex process requiring a great deal of information such as part setup, probe selection, identification of touch points, etc. This information collected through design data is used to create CAIP which produces an inspection plan table for the part. This inspection plan table can then be transferred to CMM through DMIS for the final measurement.

CMM inspection planning requires each of its steps to be designed using optimized and well-defined algorithms and techniques. This work has provided a complete system as shown in Fig. 1 that comprises of CAD, CAIP, and CMM sub-systems to perform inspection tasks effectively. Methodologies, techniques, and algorithms have been proposed for each step to finally generate an inspection plan table. Proposed methodologies and techniques have been developed for manufacturing parts created using solid modeling packages such as AUTOCAD inventor. In fact, this system takes neutral files in IGES or STEP format as input and translates the information into manufacturing information. The boundary (B-rep) geometrical information of the part design is then analyzed by a feature recognition program that has been created specifically to recognize different features on the part such as steps, holes, etc. This design information has then been used to define different steps for the development of CAIP. The execution of CAIP generates an inspection plan table for the given part. This inspection table is transformed into DMIS code which is then converted to CMM machine language using post-processor for final inspection. Moreover, a case study has been described to demonstrate the applicability of the proposed system.

2 Literature review

Various systems covering different aspects of inspection planning have been developed. Legge [2] carried out a comprehensive review on the integration of CAD with CMM and demonstrated various techniques for the generation and validation of inspection plans. There are numerous tasks in CAIP that must be carried out before the inspection plan table can be exported to CMM for inspection purposes. The expert inspection task planning system by ElMaraghy et al. [3] represents one of the earliest efforts to generate an inspection plan. This system was developed in PROLOG using feature-oriented computer-aided modeling methods. Techniques such as syntactic pattern recognition, artificial intelligence, planning rules, and logics are utilized to perform different tasks for the generation of the inspection plan. According to Spitz et al. [4], inspection planning for CMM can broadly be divided between high-level planning and low-level planning. Tasks such as workpiece setup, probe selection and orientation, etc. constitute high-level planning while the selection of measuring points, probe path generation, probe execution, etc. are classified as low-level planning. Similarly, Lee et al. [5] developed a CAIP system for on machine measurement describing two stages including global inspection planning and local inspection planning in order to allow the generation of inspection plan to be used on CMM. Global inspection planning consists of activities such as the sequence of setups and form features for the part while determination and sequencing of measuring points on the surface features constitute local inspection planning. An object-oriented planner developed by Beg et al. [6] for the inspection of prismatic parts is comprised of stages such as feature recognition, number and distribution of measuring points, accessibility analysis, probe orientations, sequencing of measuring features, etc. In this work, various algorithms and techniques, such as fuzzy logic, have been successfully applied to accomplish different tasks for the inspection plan. Roy et al. [7] developed a prototype system providing a decision planner for the automated dimensional inspection. The system was capable of extracting required information from the geometric model thus generating data for inspection planning and identifying cost-effective inspection sequences, etc. The hybrid knowledge-

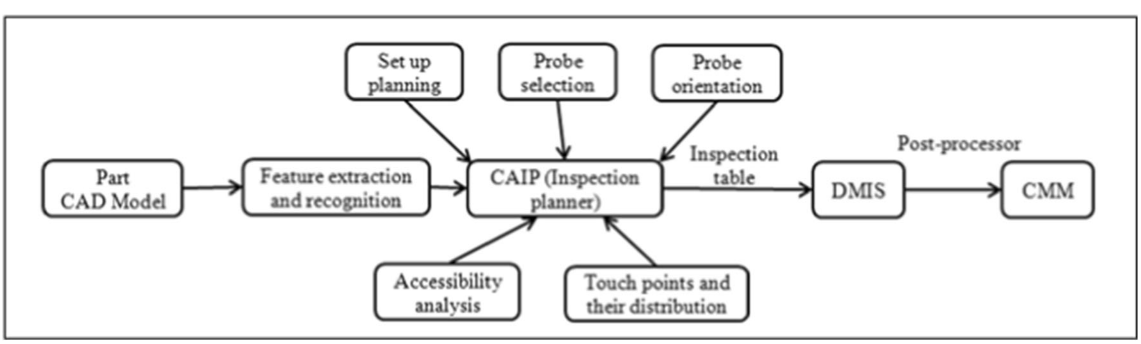


Fig. 1 Framework of CAD, CAIP, and CMM integration

based system proposed by Ghaleb et al, [8] integrated CAD and CAIP systems for the dimensional inspection of prismatic parts. This system was primarily made up of three components: design by feature module, feature recognition module, and inspection planning module. It implemented inspection knowledge and developed rules and algorithms for probe and workpiece orientation, probing point density, and distribution, collision detection and avoidance, generation of DMIS program, etc. Well-established techniques, methodologies, and algorithms have to be implemented to fulfill different tasks in the development of inspection plan. For instance, Hwang et al. [9] applied greedy heuristic method to minimize the number of part setups and probe changes and adopted Hopfield neural network for feature sequencing. Ziemian et al. [10] attended to the issues associated with part setups and probe selection in inspection planning. Approaches such as feature accessibility algorithm and geometric projection technique were utilized to outline accessible regions for each inspection feature. They also applied heuristic technique to define a set of workpiece orientations on CMM. Swept volume analysis of the probe path has been implemented by Fan et al. [11] to detect all possible collisions and generate new detoured path. Visibility map methodology introduced by Kweon et al. [12] can be used to determine part orientation on CMM. In fact, this technique provided all accessible directions through which measurement should be performed for the given part. In the same way, Corrigall et al. [13] proposed a method based on probe approach directions (PADs) to determine how the probe and the component should be set up on the machine. Moreover, techniques such as fuzzy set theory, Hammersley's algorithm, Z-map, etc., have been implemented by Cho et al. [14] to determine number of measuring points, their locations, and optimum probing paths as well as possible collisions during the inspection run. Zhang et al. [15] described inspection process planning system to produce an inspection process plan directly from the CAD model. This system comprised of five basic modules: tolerance feature analysis, accessibility analysis, clustering algorithm, path generation, and inspection process simulation to finally generate inspection plan for CMM. Vafaeesefat et al. [16] presented a methodology based on CAD model and tolerance information to establish accessibility domain for measurement points. Ajmal et al. [17] developed a knowledge-based clustering algorithm for probe selection in inspection process planning. This technique was based on grouping inspection features into feature families and probe orientations into probe cells. Application of this approach resulted in reduction of probe calibration errors, part installation errors, probe exchange time, as well as reinstallation time of parts. In this approach, incidence matrix has been developed to represent relationship between inspection features and their relative probe orientations. Furthermore, mathematical models can also be utilized to obtain appropriate probe orientations for the measurement of a point on the given feature [18]. Algorithms based on Gaussian images and Minkowski (sweeping) operations have been proposed by Spyridi et al. [19] for computing accessibility cones to minimize probe directions for inspecting a part. Generation of an efficient and collision-free path for CMM has been a crucial step in the development of CAIP and hence inspection process on CMM [20]. Therefore, algorithm such as ray tracing technique has been successfully utilized by Lin et al. [21] to generate optimal collision path for inspection on CMM. Wu et al. [22] studied probing accessibility during dimensional inspection on CMM. In this work, influence of configuration such as probe length and volume on accessibility has been thoroughly analyzed. CAIP also defines measurement attributes such as number of measurement points and their locations for various features. Jiang et al. [23] proposed a computer-aided feature-based statistical concept to determine sufficient measurement points for features.

Feature extraction has recently gained lot of attention due to its great impact on the effectiveness of CAIP and hence CMM inspection [24, 25]. It has been identified that geometric feature recognition (GFR) is an essential requirement for CAIP [26]. Lee et al. [27] proposed an integrated geometric modeling system to carry out feature-based modeling and feature recognition. This approach utilized both feature information and geometric information for feature extraction. Moreover, it handled feature interactions and protrusion features effectively by combining the capabilities of both feature-based design and feature recognition. It is very important to extract sufficient information from the CAD system in order to generate all information needed for downstream activities such as inspection. Feature extraction and recognition has been considered an important link in the development of CAIP for CMMs. Information including high-level features such as slot, pocket, hole, boss, rib, etc. and low-level entities comprising edges and vertices has to be recognized and extracted effectively. The extracted data should provide all necessary information required for the generation of inspection plan which needs to be sent to CMM. Nasr et al. [28] developed a feature recognition system based on intelligent feature recognition methodology (IFRM) in order to communicate with different CAD/CAM systems. Feature recognition processor by Li et al. [29] utilized design features and interactions of volumetric features to recognize manufacturing features from complex parts. This processor could recognize not only essential manufacturing features but also replicate, compound, and transition features defined in STEP. Sunil et al. [30] proposed a hybrid approach for recognizing interacting features from B-Rep CAD models of prismatic machined parts. Additionally, this work presented a concept of Base Explicit Feature Graphs and No-base Explicit Feature Graphs to delineate between features having planar base face like pockets, blind slots, etc. and those without planar base faces like passages, 3D features, conical bottom features, etc. Modular modeling method by Tseng [31] combined feature recognition and feature-based design in order to integrate design and process planning. In this work, they divided the part into many isolated sections based on functional and geometric analysis. In fact, this approach consisted of four stages: module creation stage, modular feature recognition stage, design with modules stage, and process planning preparation stage, in order to facilitate automated inspection. Fu et al. [32] proposed a multiple-level feature taxonomy and hierarchy based on the characteristics of part geometry and topology entities to carry out feature identification and extraction. Similarly, Bespalov et al. [33] used scale–space feature extraction technique to extract features from mechanical artifacts. With this technique, features invariant with respect to global structure of the model as well as small perturbations introduced through 3D laser scanning process could successfully be extracted. Hunter et al. [34] have also presented a functional tolerance model based on technological and topological related surfaces (TTRS) and technologic product specification (TPS) methodologies for the integration of design concepts into inspection system. This unique system was capable of extracting information related to GD&T and geometry of the part in common model. This model provided complete definition and representation of entities, attributes, and relationship of design and inspection system.

It is known that dimensional inspection has not been thoroughly analyzed as a complete system. Therefore CMM inspection has to be carried out through a sequence of operations including part setup, probe orientation, accessibility analysis, determination and distribution of touch points, etc. There is a need for a comprehensive framework representing feature extraction and recognition module, CAIP, and CMM inspection in a seamless and integrated fashion. It has also been observed in the literature that effectiveness of measurement process on CMM greatly depends on the appropriateness of information provided in CAIP. A variety of algorithms, tools, methodologies have been developed in this work to enhance the working of CAIP. Existing works have taken up and studied various issues associated with different processes, but very little work representing complete systems in an integrated manner have been found. In this work, an attempt has been made to integrate the feature recognition module, CAIP, and its various inspection operations, generation of DMIS code based on the inspection plan table and finally CMM inspection. A variety of algorithms and techniques have been deployed to achieve the objectives of reliable and effective inspection on CMM. In fact, this research has focused on various techniques and algorithms required to improve each stage of CAIP.

3 Proposed system

Proposed system represents a generative inspection planning system that has been developed using both Initial Graphics Exchange Specification (IGES) and Standard for the Exchange of Product (STEP)-based modeling environment. The system retrieves inspection-related information including dimensions, tolerances, etc. using either IGES or STEP model libraries to define different stages for CAIP. CAIP consists of various processes such as setup planning, probe orientation, determination of measurement points, collision avoidance, etc. The corresponding results in the form of the inspection plan table from CAIP are converted into DMIS programs. DMIS programs are then transformed to CMM machine language using post-processor. This system allows the user to recognize high-level features such as slot, pocket, hole, etc., and lower-level entities, such as edge, vertices, etc., for inspection.

The presented system as shown in Fig. 2 contains three basic components which include the CAD module or feature extraction and recognition module, CAIP, and CMM. This system has integrated CAD and CAIP systems in order to direct the CMM for dimensional inspection of manufacturing parts. In this work, CAIP has been generated based on the geometric information obtained from CAD. To begin with, all desired design information that exist in the CAD model have to be transformed to either IGES or STEP formats depending on the compatibility of the CAD software. These STEP or IGES files have then been used for feature extraction and recognition using object-oriented structures (such as C++). After retrieving relevant design information, activities such as setup planning, probe selection and orientation, accessibility analysis, determination of measurement points, etc. have been performed to generate CAIP. The output of the CAIP module is an inspection plan table that goes into last module, i.e., CMM module for final inspection. Before the execution of CMM takes place, the inspection table has to be converted to a DMIS program which is then transformed into CMM machine language with the help of a post-processor. The different stages of the proposed system have been elaborately described in the following sections.

3.1 Feature extraction and recognition

Feature extraction and recognition has been an essential requirement for the development of the CAIP and the CMM inspection. Different issues associated with feature extraction and recognition have attracted a great deal of attention over the last few decades. The CAD model is an important entity and needs to be interpreted for feature extraction and recognition. Although CAD files contain detailed geometric information of the part, they are not suitable for use in downstream applications that include process planning. A feature recognition methodology is needed to develop a feature recognition system that can effectively communicate with CAD/CAM systems. Therefore, the methodology required to perform the feature extraction and recognition has been carried out.

Feature extraction and recognition methodology commences with the introduction of a part design from CAD software. The part design is represented in the form of a solid model created using constructive solid geometry (CSG) technique. Geometrical information (e.g., lines, faces, vertices, etc.) of the part design has been obtained in the form of either an IGES or STEP file. This information of part design has then been analyzed by the feature recognition program to extract

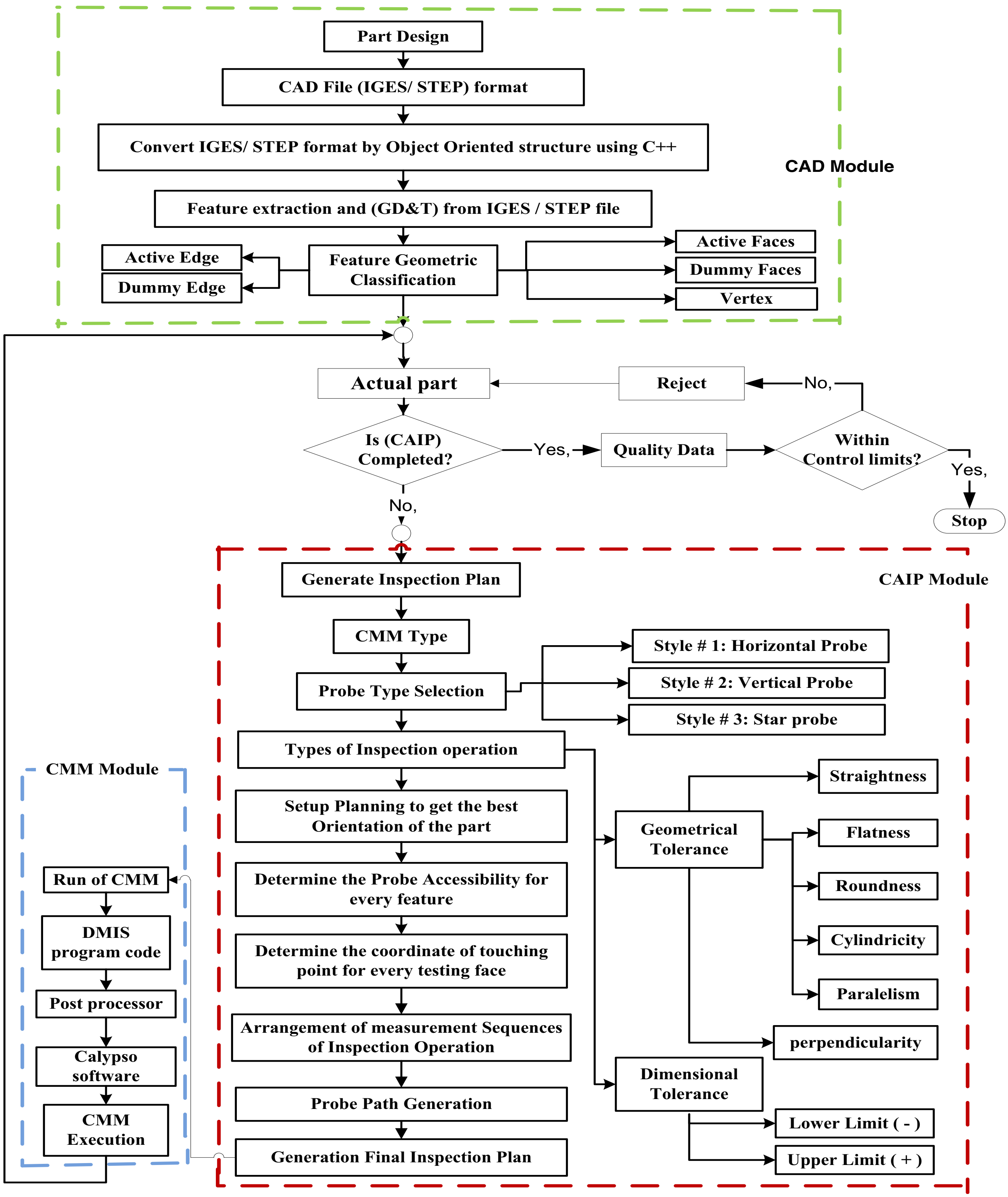


Fig. 2 Proposed system to carry out inspection

different features based on geometric reasoning and object-oriented approaches. The feature recognition program can recognize features such as slots (through, blind, and round comers), pockets (through, blind, and round corners), inclined surfaces, holes (blind and through) and steps (through, blind, and round corners), etc. These features which represent manufacturing information can then be mapped to process planning as an application for CAM/CAIP.

The structure of feature extraction and recognition methodology [35] adopted in this work has been shown in Fig. 3. It consists of following three main phases:

- Data file converter phase which converts CAD information in STEP (or IGES) format into object-oriented structure
- Object form feature classifier phases to classify part geometric features into different feature groups
- Manufacturing features classifier maps the extracted features (manufacturing information) to process planning using production rules

Following are the proposed steps for features extraction and classifications:

Step 1 Extract geometry and topology entities for the designed object model from IGES file or STEP file format
Step 2 Extract topology entities in each basic surface and identify its type
Step 3 Test the feature's existence in the basic surface based on loops
Step 4 Identify feature type
Step 5 Identify the detailed features and extract the related feature geometry parameters
Step 6 Extract all GD&T test faces depending on the functionality of the part
Step 7 Identify detailed of the process plan for each feature

3.1.1 Feature extraction using IGES

Mechanical Desktop 6 power pack® CAD has been used for part design which supports the IGES file format translation (B-REP Solid (186) with Analytical Surfaces). The feature recognition program has been developed using Windows-based Microsoft Visual C++ 6 on a PC environment. Extracted entities include vertices, edges, loops, and faces while feature recognition involves identification and grouping of feature entities from geometric models [35]. GD&T has been extracted from CAD model and then exported into IGES translator and treated as a note in the exported IGES file. The file includes all datum faces, geometrical tolerance test type, its value, boundary of faces for every test, and its datum.

GD&T can be extracted from the CAD model exported in the IGES translator. IGES does not support tolerances but represents them as a note in the exported IGES file. The general note which translates GD&T in the IGES file format in the parameter data of the IGES structure has been presented in Fig. 4.

The symbols for the GD&T used in IGES format have been listed in Table 1. The first step in GD&T extraction has been achieved by redefining IGES in the form of the object-oriented data structure as demonstrated in Fig. 5. Algorithm for datum extraction and test extraction can be seen in Figs. 6 and 7, respectively.

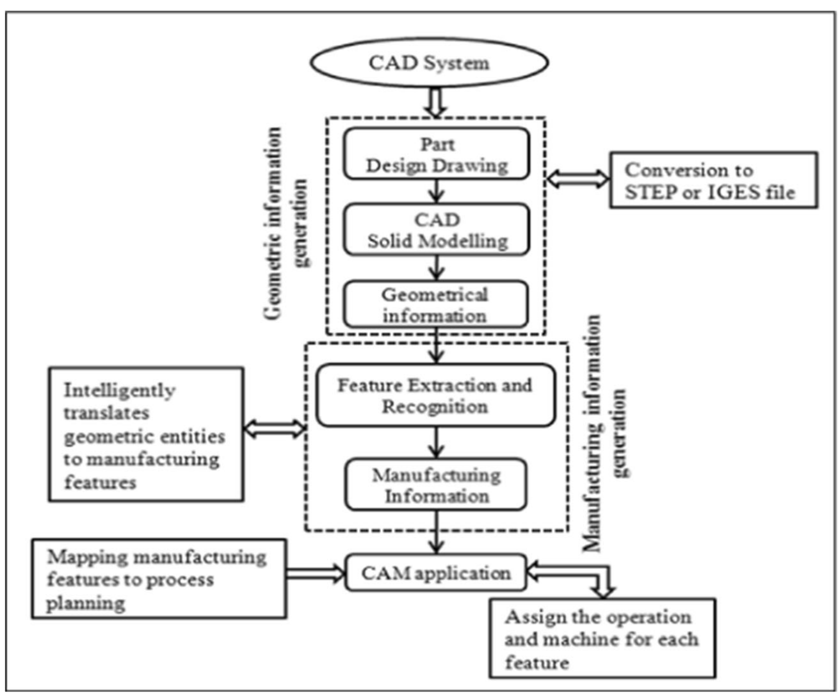


Fig. 3 Feature extraction and recognition methodology [35]

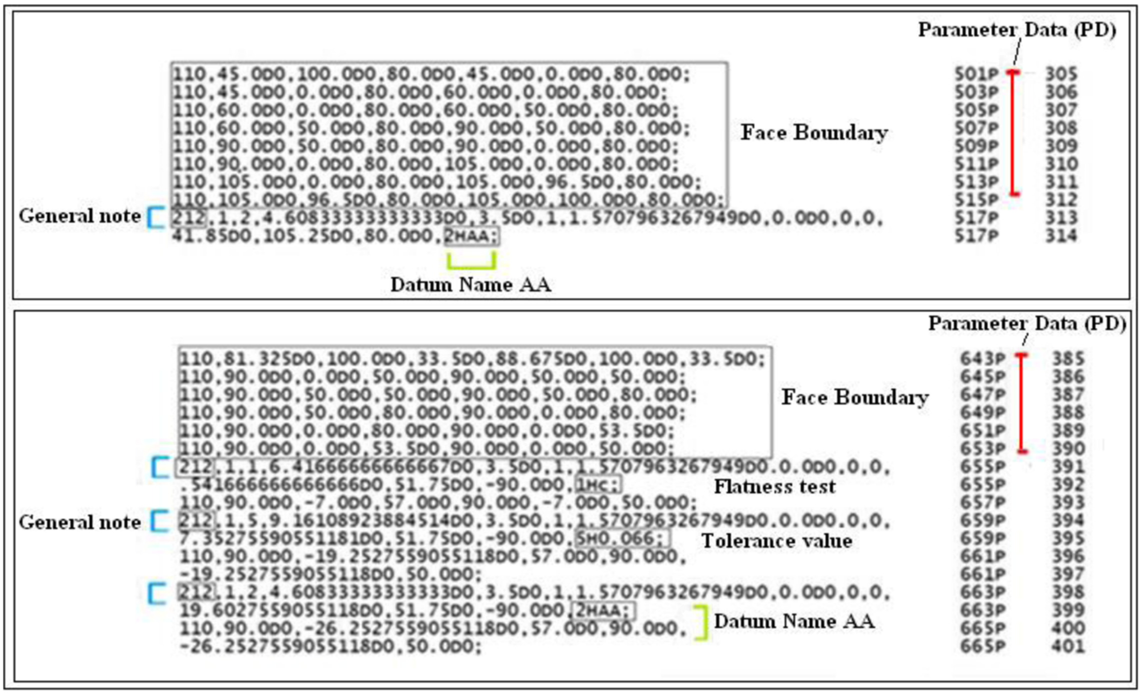


Fig. 4 Structure of the GD&T in the IGES file format

In this diagram, the IGES class is the number of annotation. In fact, the IGES class consists of the following procedures: get P-Entry character, get test name, get test boundary, get test type, get test value, get test datum, and make order_point (inspection) file. The points that have been used for these methods are extracted as follows: getting the boundary in the IGES file by getting the points (X, Y, Z). Feature extraction and recognition class consists of methods such as getting the

Table 1 Library of general note of GD&T symbol in IGES file format

IGES Symbol	Tolerance Type	Test Symbol	Test Type
Hu	Tolerance, single element	—	Straightness
Hc		⏥	Flatness
He		○	Circularity
Hg		⌭	Cylindricity
Hk		⌒	For line shape
Hd		⌓	For surface shape
Hf	Tolerance guidance for other elements	//	Parallelism
Hb		⊥	Perpendicularity
Ha		∠	Inclination (slope)
Hj	Tolerance of position relative to other elements	⌖	Position
Hr		◎	Concentricity
Hi		⌯	Symmetry

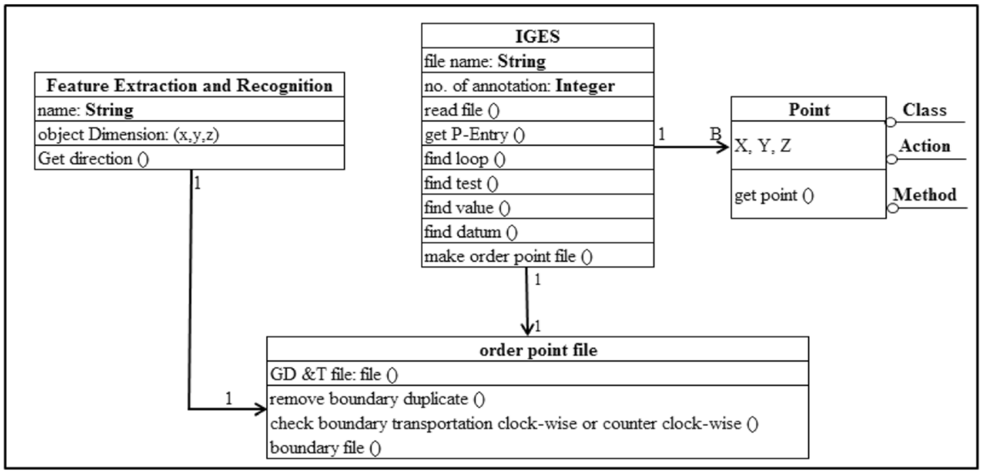


Fig. 5 OOP Class diagram for extraction of GD&T in IGES file

direction of GD&T. The order_point class entails removing duplicate boundaries and checking boundary point's transportation (clockwise or counter clockwise) by reading the boundary file.

The output of feature extraction and recognition and GD&T from the IGES file format is shown in Fig. 8a and b, respectively.

3.1.2 GD&T extraction using STEP

A proposed methodology has been developed for 3-D manufacturing parts that are created using CATIA V5 R21. This system places a file in STEP (identified as ISO 10303) as input and then translates the information from the file into manufacturing information. STEP file format has been exported at AP203 which deals with configuration controlled 3D designs of mechanical parts and assemblies (ISO10303-203:1994), one of the most widely used application protocols of STEP [36]. Furthermore, the feature recognition program has been developed using Windows-based Microsoft Visual C++ 6 on a PC environment. AP-203 Edition 2 is a recently released new version of the AP-203 standard for exchanging 3D geometry between CAD systems and one of the extensions which includes GD&T data [36, 37].

STEP is one of the formats that attaches tolerance information onto the various features of the part. It generally associates tolerance entities with shape_aspect in order to identify the tolerance feature. The part features for numerous cases of the solid boundary representation model are mainly represented by advanced_face entities in STEP. For example, a through hole in a solid model can be represented by the advanced_face entity such as the semi-circular surface. The shape_representation which can be defined as the representation of the shape_aspect for the feature is typically a connected_face_set and exhibits the same geometric_representation_context as the solid. The topological_representation_items are collected together by a shape_representation in STEP as shown in Fig. 9.

Application of composite_shape_aspect is used rather than shape_aspect in the cases where tolerance needs to be applied

Fig. 6 Algorithm for feature extraction

Step 1: Open IGES file

Step 2: Create PSection file (container to store PEntry objects)

Step 3: While Read line from PSection file

- 3.1. If there are more than 4 line of 110 Pentry followed by 212 line
  - 3.1.1. If the last value contains 2H
    - 3.1.1.1. Take as datum name
    - 3.1.1.2. Take the last point and Go back until get atleast two matched points
    - 3.1.1.3. Set the points which we get it from previous step as a datum boundary
  - 3.1.2. End If loop
- 3.2. End If loop

Step 4: End of while loop

Fig. 7 Algorithm for test extraction

```
Step 1: Open IGES file

Step 2: Create PSection file (container to store PEntry objects)

Step 3: While Read line from PSection file

    3.1. If there are more than 4 line of 110 Pentry followed by 212 line
        3.1.1. If the last value contains 1H
            3.1.1.1. Read the character after 1H set his value from library (table 1)
                3.1.1.1.1. Take it as test name
            3.1.1.2. While read line 212 and if we do not get before this line more than 3 times 110 line
                3.1.1.2.1. Extract the value of the test which will be in the last field
            3.1.1.3. End of while loop
        3.1.2. End of If loop
    3.2. End of If loop
Step 4: End of while loop
```

to more than one feature such as pattern of holes. Moreover, the algorithm for extraction of GD&T using STEP has been described below while flow chart for extraction can be seen in Fig. 10.

```
1. Read STEP file
2. Read lines until lines contain "VALUE_REPRESENTATION_ITEM ('number of annotations')"
3. Find number of annotation N
4. For J=1 to N
   4.1 Give name (J)
   4.1.1 Open STEP file
   4.1.2 Read lines until line contain "DRAUGHTING_MODEL_ITEM_ASSOCIATION"
   4.1.3 Find the name in the line
   4.1.4 Print Name
                4.2 Read boundary (J)
                    4.2.1 Open STEP file
                    4.2.2 Read lines until line contains
                        "GEOMETRIC_ITEM_SPECIFIC_USAGE"
                  4.2.3 Find the "ADVANCED_FACE"
                        4.2.4 Use "ADVANCED_FACE" to find face outer
                        4.2.5 Use "ADVANCED_FACE" to find edge loop
                4.3 Read type (J)
                       4.3.1 Open STEP file
                4.3.2 Read lines until line contain "GEOMETRIC_CURVE_SET"
                        4.3.3 Find type
                        4.3.4 Print type
  5. End
```

There are specific production rules that define how features should be extracted. For example, the algorithm shown in Fig. 11 can be used for the extraction of the slot blind feature shown in Fig. 12.

Extraction of GD&T information from STEP by redefining it in the form of the object-oriented data structure can be demonstrated in Fig. 13. In this diagram, STEP class is the annotation number. STEP class consists of the following procedures: obtain number of annotation, extract G&DT and make order_point (inspection) file. GD&T class is used for extracting GD&T which consists of methods that include get test name, get test boundary, get test type, get test value, get test datum, and get test direction. The order_point class contains the following methods: remove the duplicate boundary, check boundary point's transportation (clockwise or counter clockwise) by reading the boundary file.

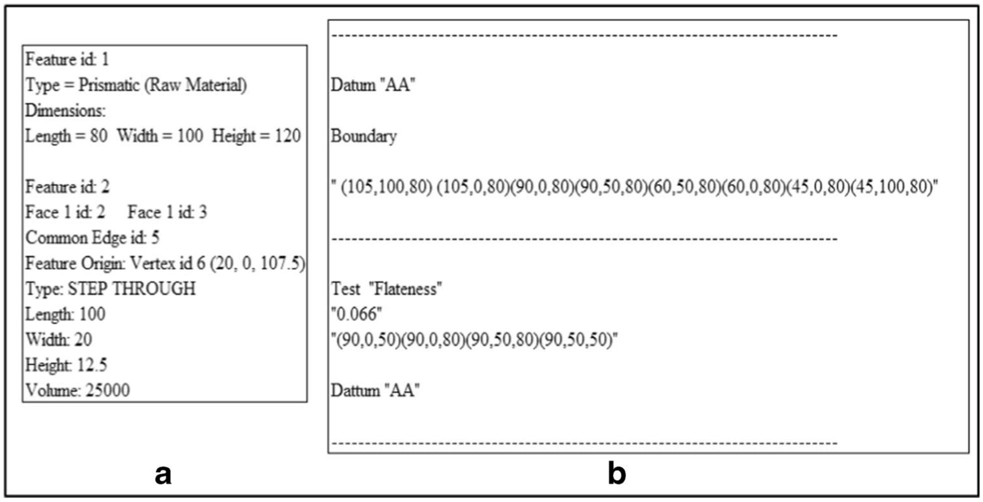


Fig. 8 Output of (**a**) feature extraction and recognition and (**b**) GD&T from IGES file format

3.2 Accessibility analysis

Accessibility analysis has been an important aspect in the generation of the inspection plan table for CMM [22]. It significantly reduces the number of unnecessary changes in probe orientation and maximizes the number of features inspected using the same probe orientation. In this section, all accessible probe orientations for every surface feature have been carefully evaluated. Clustering the algorithm groups, the inspection probe and surface features, into inspection group allows the time for the inspection probe exchange and calibration to be reduced to a minimum.

3.2.1 Probe accessibility direction

PAD represents the accessibility direction of the probe as it touches each feature and the direction for individual or group features. The first step in the clustering algorithm is to ensure that the clustering features have the same PAD that can be inspected in one operation. PADs for different features have been shown in Fig. 14.

After analyzing the feasibility of PAD for each feature, the features of a PAD matrix can be constructed. Being a part with m features that can be inspected using n PADs based on the inspection probe, PAD matrix is a $m \times n$ matrix as shown in Fig. 15.

Where

m represents the number of rows in the PAD matrix i.e. number of features

n represents the number of columns in the PAD matrix i.e. number of probe orientations

If feature fj has PADi, then rij=1, otherwise rij=0

The determination of probe accessibility analysis for every feature has been generated automatically.

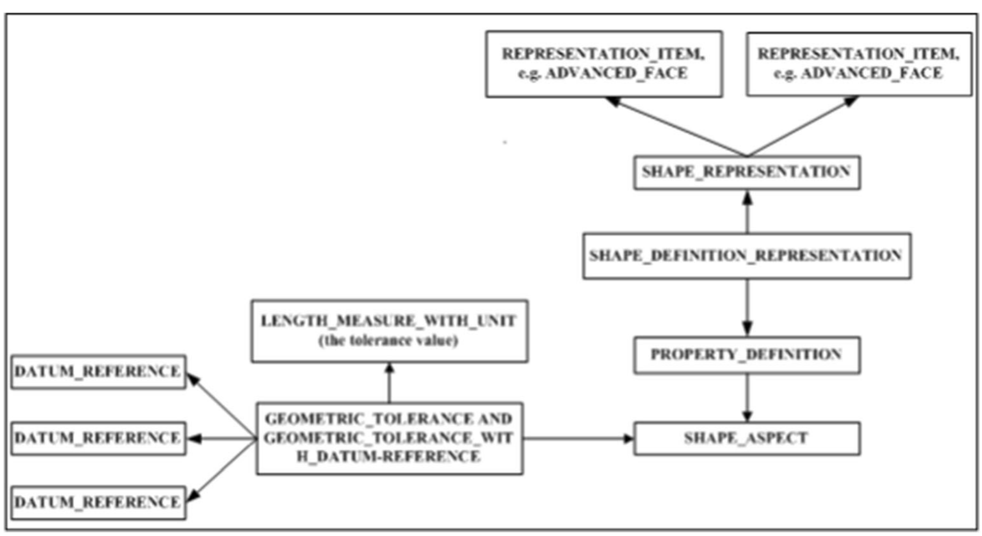


Fig. 9 Relationship of tolerance entities and shape elements

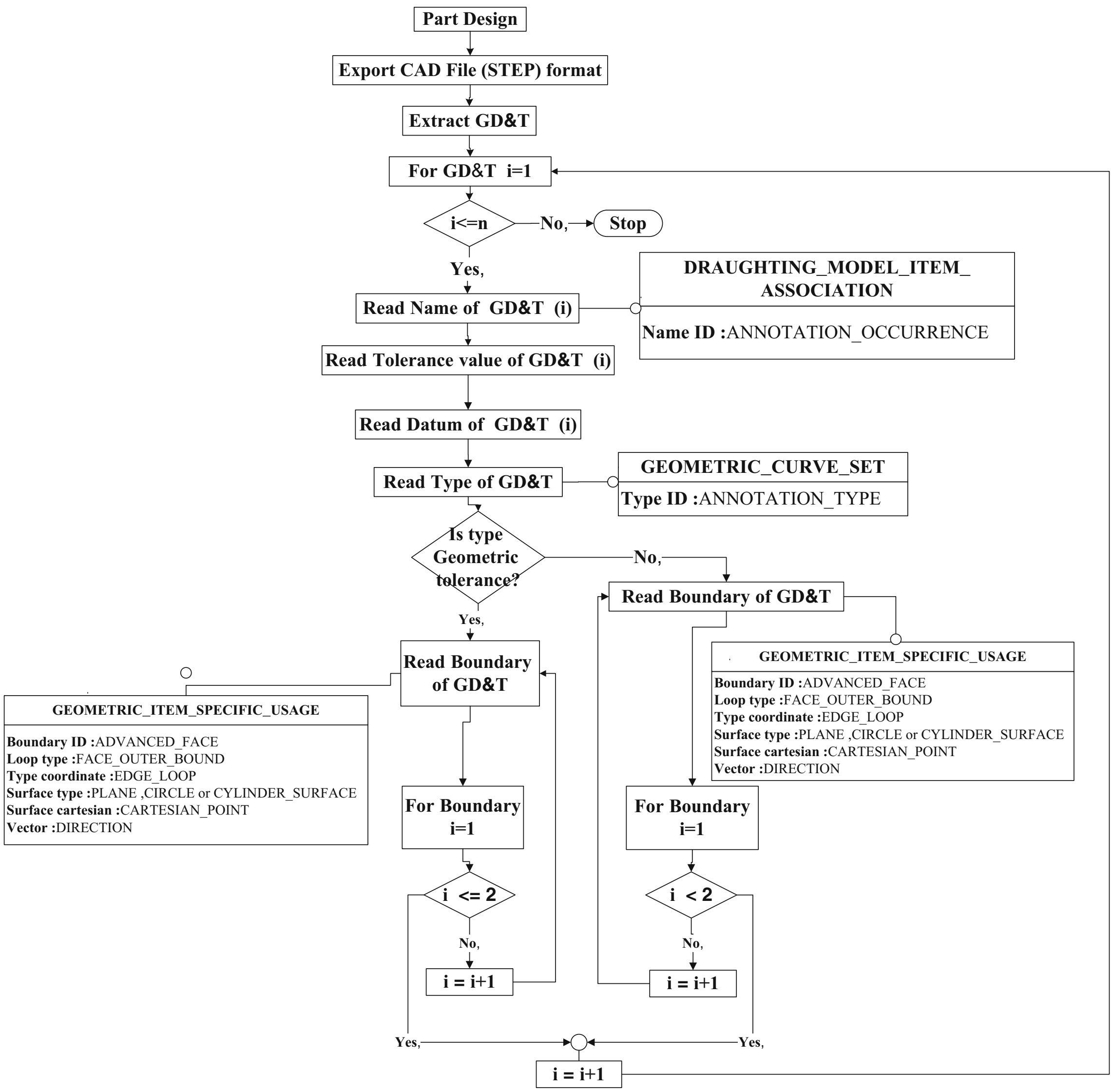


Fig. 10 Flow chart representing feature extraction using STEP

3.2.2 Approach direction depth

Approach Direction Depth (ADD) of a feature can be defined as the depth accessibility of the probe during inspection. The ADD is measured from the highest point on a part to the lowest point in the feature with the part orientated in PAD. A slot through feature, as shown in shown in Fig. 16, can be measured using PAD_1, PAD_2, and PAD_3 with ADD of ADD_1, ADD_2, and ADD_3, respectively.

After calculating the feasible ADD_s for each feature, a feature ADD matrix can be constructed. For a part with m features that can be inspected using n PAD_s, an ADD matrix is an $m \times n$ matrix as shown in Fig. 17.

Fig. 11 Algorithm for extraction of slot blind feature

Step 1: For every four faces of surface type plane (named as face1 to face4).
Step 2: If face1, face2 concave edge count of outer loop equals 3 each and face3, face4 concave edge count of outer loop equals 2 each, and face1 connected to face2 by one edge.
    2.1. If face1 perpendicular to face2, face 3, and face 4 also face 3 and face4 parallel to each other.
        2.2.1. SLOT BLIND found
        2.2.2. Create a new SlotB object and add to feature list.
Step 3: End For

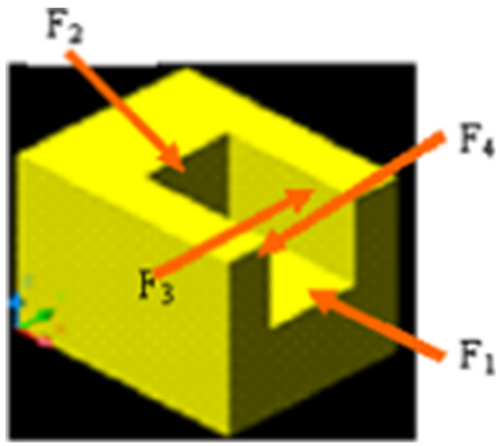


Fig. 12 Slot blind feature

where

m represents the number of rows in the matrix, i.e., number of features in the part n represents the number of columns in the matrix, i.e., number of probe orientations gij represents ADD of feature f_j

From the PAD matrix, a probe type will be selected depending on the features covered at a given setup. ADD will give some information required in the probe selection such as the length of the probe, diameter of the probe sphere, etc. The algorithm for PAD analysis has been described as follows:

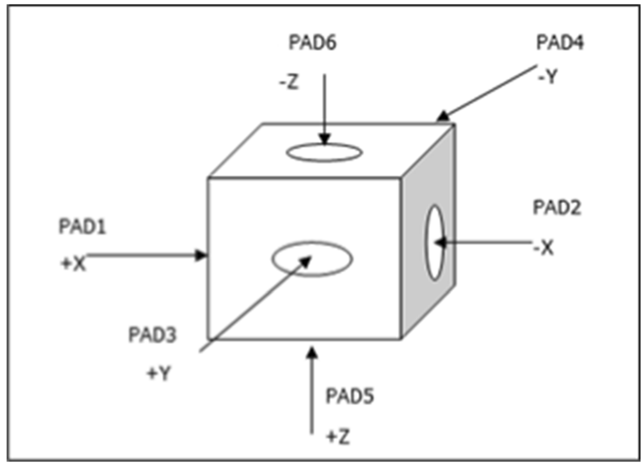


Fig. 14 Probe accessibility directions (PADs)

```
Step 1: Open Setup file
Step 2: While file is not end then read line by
        line for each feature
Step 3: Read all faces in the feature and nor-
        mal vector
Step 4: Create matrix contains six columns (+
        x, −x, +y, −y, +z, −z)
```

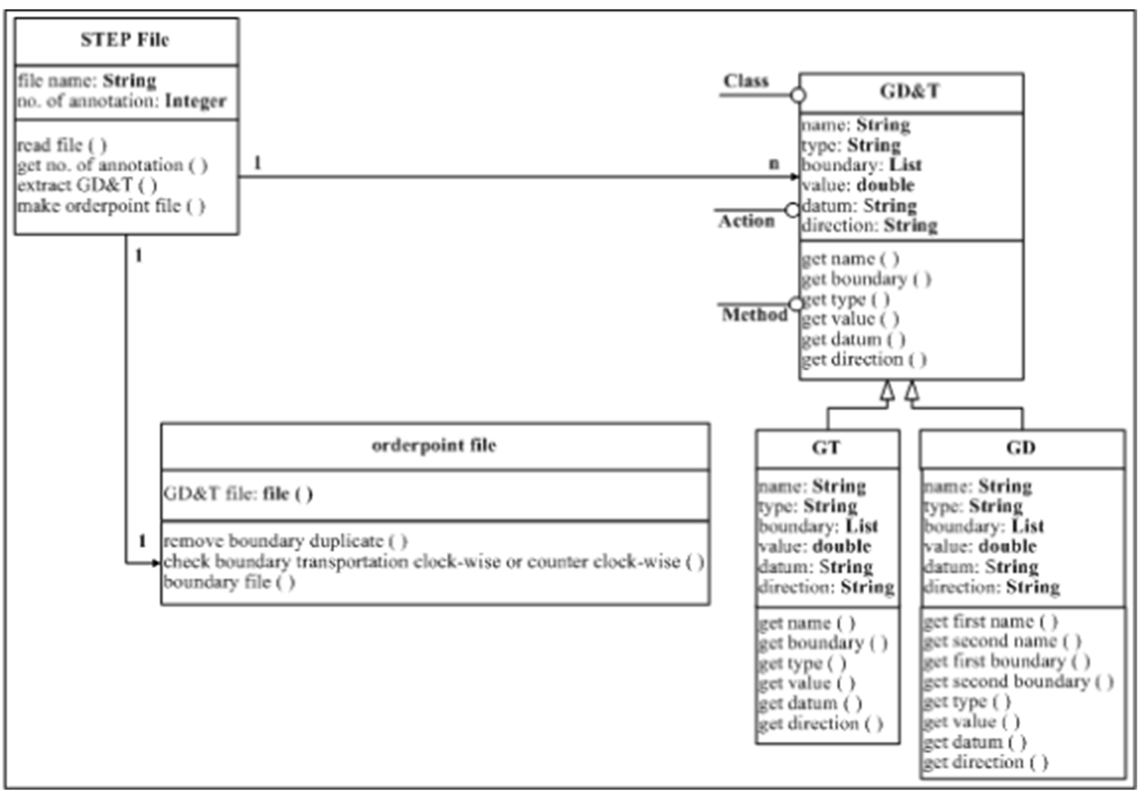


Fig. 13 OOP class diagram of extraction GD&T using STEP

	PAD1	PAD2	PAD3	PAD4	PAD5	....	PADn
f1	r11	r12	r13	r14	r15	....	r1n
f2	r21	r22	r23	r24	r25	....	r2n
f3	r31	r32	r33	r34	r35	....	r3n
f4	r41	r42	r43	r44	r45	....	r4n
⋮	⋮	⋮	⋮	⋮	⋮	....	⋮
fm	rm1	rm2	rm3	rm4	rm5	....	rmn

Fig. 15 PAD ($m \times n$) matrix

```
Step 5:  Store the first element from normal
         vector in columns (+x, −x) based on
         the sign of the element
Step 6:  Store the second element from normal
         vector in columns (+y, −y) based on
         the sign of the element
Step 7:  Store the third element from normal
         vector in columns (+z, −z) based on
         the sign of the element
Step 8:  Compute the summation of every column
         in the matrix
Step 9:  If summation is greater than zero
           Set it = 0
            Else
           Set it = 1
Step 10: End if loop
```

For example, the blind slot, shown in Fig. 18, has dimensions of 20, 60, and 20 mm length, width, and height, respectively. It possesses only two probe directions.

The normal vector for each face of the feature has been used to find the accessibility of the probe in the slot blind feature as shown in Table 2. PAD and ADD matrices for blind slot have been shown in Figs. 19 and 20, respectively.

PADs for slot blind feature are $+y$ and $-z$ directions within ADDs 60 and 20, respectively, as shown in the matrices. This information from the PAD and ADD matrices has to be used to select the best probe (considering the length of the probe, diameter of the probe sphere, etc.) for a given feature.

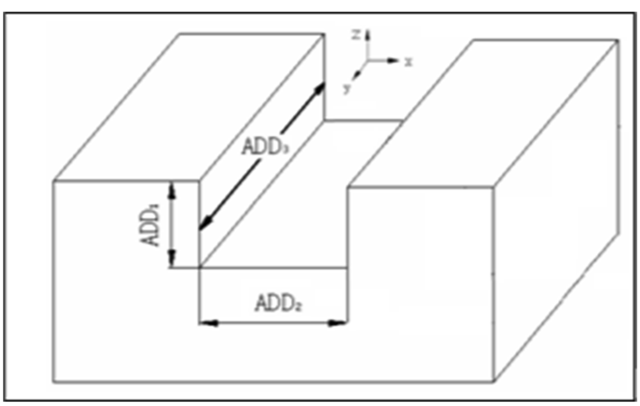


Fig. 16 Slot through feature

	PAD1	PAD2	PAD3	PAD4	PAD5	....	PADn
f1	g11	g12	g13	g14	g15	....	g1n
f2	g21	g22	g23	g24	g25	....	g2n
f3	g31	g32	g33	g34	g35	....	g3n
f4	g41	g42	g43	g44	g45	....	g4n
⋮	⋮	⋮	⋮	⋮	⋮	....	⋮
fm	gm1	gm2	gm3	gm4	gm5	....	gmn

Fig. 17 ADD matrix

3.3 Setup planning

Measuring part often requires more than one orientation during inspection on CMM. Therefore, it is very critical to identify part orientations so that all tolerances and dimensions can successfully be inspected with the minimum number of part setups. It has also been discovered that feature accessibility is greatly affected by inspection feature's orientation, location, dimension, and its interactions with other features.

Setup planning determines how part should be oriented on CMM machine table so that the maximum number of features can be measured in one setup. It becomes crucial especially when the time needed to change part setup is significant with respect to overall inspection time of part. This section of the paper has focused on determining the best part setup for an automated inspection plan. Two different approaches have been proposed and implemented for part setup planning. The main idea for setup planning is to identify part face which has a minimum number of inspecting features. It determines the face that would result in the inspection of maximum possible features. This face determines base face for part orientation on machine table. Base face is referred to as preferential base or primary locating face in setup planning.

3.3.1 First rule (numerical method)

This method makes use of artificial neural network (ANN) to predict the best setup. Geometric extracting entities and features with the same PAD are identified as input to ANN. In

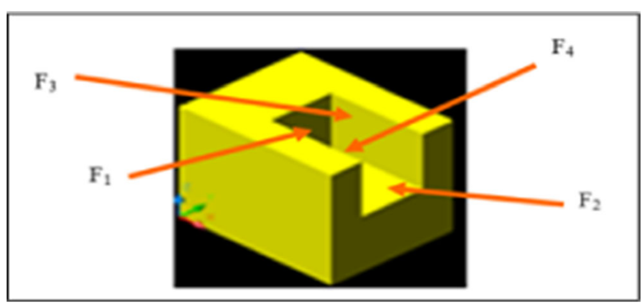


Fig. 18 Blind slot having only two probe directions

Table 2 Normal vector of each face in the feature

	Normal vector	(+*X*)	(−*X*)	(+*Y*)	(−*Y*)	(+*Z*)	(−*Z*)
Face ID no. 1	(−1,0,0)	0	1	0	0	0	0
Face ID no. 2	(1,0,0)	1	0	0	0	0	0
Face ID no. 3	(0,–1,0)	0	0	0	1	0	0
Face ID no. 4	(0,0,1)	0	0	0	0	1	0
Sum		1	1	0	1	1	0
Accessibility		0	0	1	0	0	1

order to implement this approach for given problem, inputs to ANN are determined as follows:

Geometric extracting entities It includes geometric entities such as the number of vertices, line edges, circular edges, internal loop, external loop, concave faces, convex faces, etc. All these entities are extracted from the extraction and recognition file.

Features having same PAD Features with the same PAD were determined as follows:

- Six PADs (+*x*, −*x*, +*y*, −*y*, +*z*, and –*z*) and six setups (S (right), S (left), S (front), S (rear), S (top), S (bottom)) were identified for rectangular block
- For given setup, number of features that could be accessed for each PAD was determined and presented in form of ($m \times n$) matrix as shown in Fig. 21. In this matrix, f11 represents number of features that can be inspected with +*x* probe direction when right face of part is used as primary locating face. Similarly f54 represents number of features that can be inspected with –*y* direction when top face act as primary locating face.
- For each PAD, total number of features that could be accessed was calculated, e.g., total number of features for +*x* PAD was equal to (f11+f21+f31+f41+f51+f61). Similarly, total number of features for +z PAD was equal to (f15+f25+f35+f45+f55+f65)
- Finally, six inputs in form of summation of PAD_j in each column were obtained.

Therefore, the total of 13 inputs (six PADs + seven extracted geometries) were identified for input layer and six nodes including bottom face, top face, front face, rear face, left face, and right face were selected for output layer.

Training and testing of experiments Training experiments determine number of hidden layers using EasyNN plus to optimize network structures. Several training experiments with different numbers of hidden neurons, learning rates (0.60), and momentum values (0.80) were checked for best training parameters and minimum error.

Result of testing After several training experiments, the network was successfully trained with average validating error of 0.0025.

Once the training was successfully finished, the network was validated for ten examples with the validating percentage of 80 %.

3.3.2 Second rule (graphical method)

In this method, the best setup was determined based on the number of interacting features as shown in Fig. 22. Face with its minimum number of interactions was selected as a primary locating face.

Different steps to identify best setup for a part can be defined as follows:

- Divide all faces of the given prismatic part as primary faces and secondary faces. Primary faces for rectangular blocks are faces that determine the basic shape of the part. Top, bottom, front, rear, right, and left faces fall in to this category. Secondary faces are faces that belong to various features on part such as slot, rib, boss, pocket, etc.
- Determine the interaction between primary faces and edges of secondary faces. For example, left face f_{24} (as shown in Fig. 23a) has interaction with only 1 edge, top faces f_{27} and f_{17} (as shown in Fig. 23b) have interactions with 8 edges, right face f_{28} (as shown in Fig. 23c) has interaction with only 1 edge, bottom faces f_{12} and f_{21} (as shown in Fig. 23d) have interactions with 6 edges, front face f_7 (as shown in Fig. 23e) have interactions with 16 edges, and rear face f_{16} (as shown in Fig. 23f) have interactions with 10 edges.

The primary faces would be arranged in ascending order of number of interactions in order to determine the best setup.

- Select primary face with a minimum number of interactions as primary locating face. For example, left face f_{24} and right face f_{28} have least interactions (one interaction each). Therefore, either left face or right face can be

Fig. 19 PAD matrix for slot blind

	+x (PAD)	−x (PAD)	+y (PAD)	−y(PAD)	+z(PAD)	−z (PAD)
Slot Blind	0	0	1	0	0	1

Fig. 20 ADD matrix for slot blind

	+x (PAD)	−x (PAD)	+y (PAD)	−y(PAD)	+z(PAD)	−z (PAD)
Slot Blind	0	0	60	0	0	20

selected for primary locating face as shown in Fig. 24. These faces as primary locating face (left face f_{24} or right face f_{28}) allow probe to inspect maximum features in one setup. With this orientation, probe can inspect features with feature IDs (2, 4, 5, 6, 7, and 8).

3.4 Measurement points

It is very important to compute the number of measurement points and their coordinates on the measuring feature for the generation of inspection plan. This module determines the distribution of measurement points on the measuring feature. Different algorithms have been proposed in this section to determine the number of probing points. Methodologies for the generation of measurements for rectangular and cylindrical faces have described in following sections.

3.4.1 Touch point generation for rectangular face

In the extraction module, every boundary of the rectangular face is represented by its boundary corner and by using (x, y, z) coordinates. These coordinates are used as input for the algorithm as shown in Fig. 25. The sequence of the boundary points can be either clockwise direction or counter clockwise direction.

The new boundary is marked with blue and pink lines within the original boundaries of the rectangular face as shown in Fig. 25. The probe travels a distance (r) from the edge of the original boundary surface to the probing path that is the radius of the probe sphere. The methodology for the generation of touch point on a rectangular face can be described as follows.

Step 1 Input the number of vertices "n" for a given plane, e.g., for this plane "n=12"

Step 2 Input the radius of the probe sphere (r).

	+x(PAD)	-x(PAD)	+y(PAD)	−y(PAD)	+z(PAD)	−z(PAD)
S(Right)	f11	f12	f13	f14	f15	f16
S(Left)	f21	f22	f23	f24	f25	f26
S(Front)	f31	f32	f33	f34	f35	f36
S(Rear)	f41	f42	f43	f44	f45	f46
S(Top)	f51	f52	f53	f54	f55	f56
S(Bottom)	f61	f62	f63	f64	f65	f66

Fig. 21 Number of features for different PADs and part faces

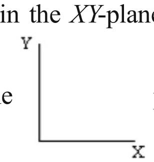
Step 3 Check which plane contains the extracted vertices as follows:

If $\{(X1 = X2) \text{ and } (Y1 = Yn)\}$ or $\{(X1 = Xn) \text{ and } (Y1 = Y2)\}$ then it is a XY-plane
If $\{(X1 = X2) \text{ and } (Z1 = Zn)\}$ or $\{(X1 = Xn) \text{ and } (Z1 = Z2)\}$ then it is a XZ-plane
If $\{(Y1 = Y2) \text{ and } (Z1 = Zn)\}$ or $\{(Y1 = Yn) \text{ and } (Z1 = Z2)\}$ then it is a YZ-plane

Step 4 Input the ball radius (r) of the stylus.

Step 5 Input the coordinates of all the vertices of the plane surface extracted from the IGES file.

$$(X1, Y1, Z1); (X2, Y2, Z2); (X3, Y3, Z3); (X4, Y4, Z4); (X5, Y5, Z5); (X6, Y6, Z6); (X7, Y7, Z7); (X8, Y8, Z8); (X9, Y9, Z9); (X10, Y10, Z10); (X11, Y11, Z11); (X12, Y12, Z12), \ldots.., (Xn, Yn, Zn).$$

Step 6 Assuming the plane surface is in the XY-plane and calculating the coordinates of the 

probing points (U, V, W)

Since, the plane is XY-plane, therefore $W1=Z1$

a. Determine if this boundary is in clockwise or counter clockwise direction. If it is in the clockwise direction

If $\sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\}$ then $Ui = (X_i + r), V_i = (Y_i - r)$
If $\sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\}$ then $Ui = (X_i + r), V_i = (Y_i + r)$
If $\sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\}$ then $Ui = (X_i + r), V_i = (Y_i + r)$
If $\sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\}$ then $Ui = (X_i + r), V_i = (Y_i - r)$

Fig. 22 Setup planning algorithm

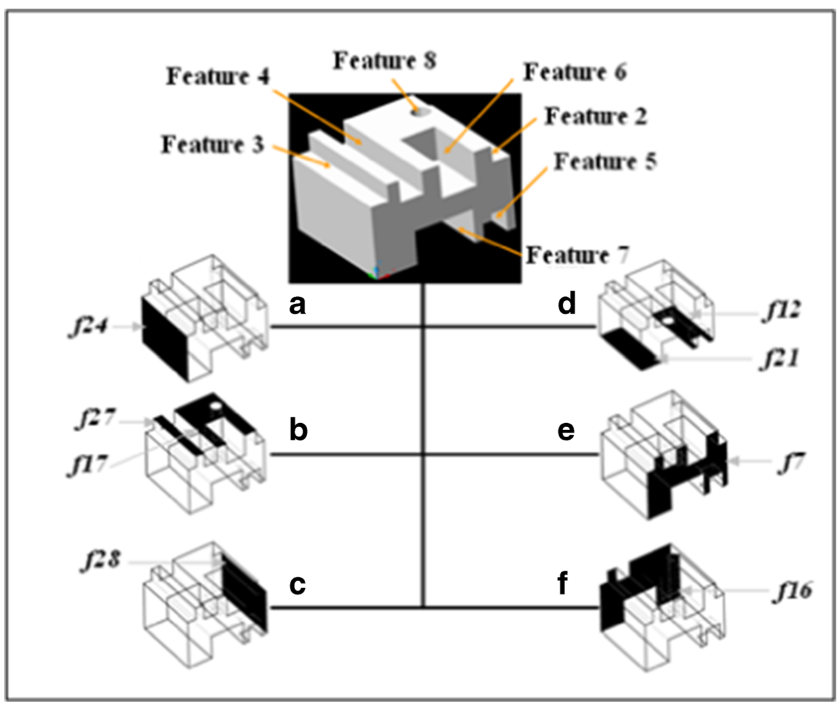


Fig. 23 Prismatic part and interactions edges

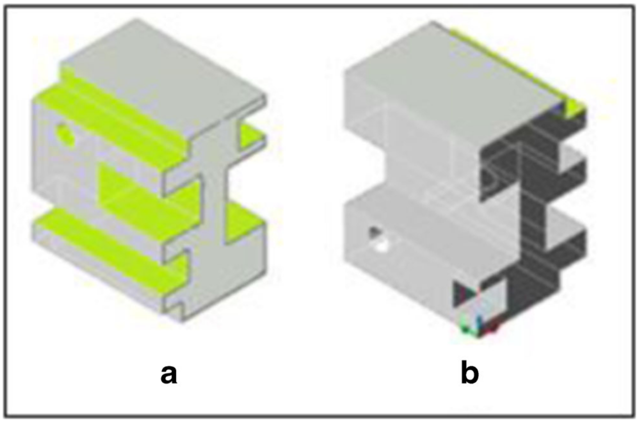


Fig. 24 Setup planning

If it is in the counter clockwise direction

$$\text{If } \sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\} \text{ then } Ui = (X_i + r), V_i = (Y_i + r)$$

$$\text{If } \sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\} \text{ then } Ui = (X_i + r), V_i = (Y_i - r)$$

$$\text{If } \sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\} \text{ then } Ui = (X_i + r), V_i = (Y_i + r)$$

$$\text{If } \sum_{i=1}^{n}\{(X_{i+1} > X_i)\&\&(Y_{i+1} = Y_i)\} \text{ then } Ui = (X_i + r), V_i = (Y_i - r)$$

b. Calculate the remaining points as follows;

$$\sum_{i=1}^{n} U_{(i+n)} = \sum_{i=1}^{n} \frac{(X_i + X_{i+1})}{2} \text{ and } \sum_{i=1}^{n} V_{(i+n)} = \sum_{i=1}^{n} \frac{(Y_i + Y_{i+1})}{2}$$

Fig. 25 Computation of coordinates for touch points

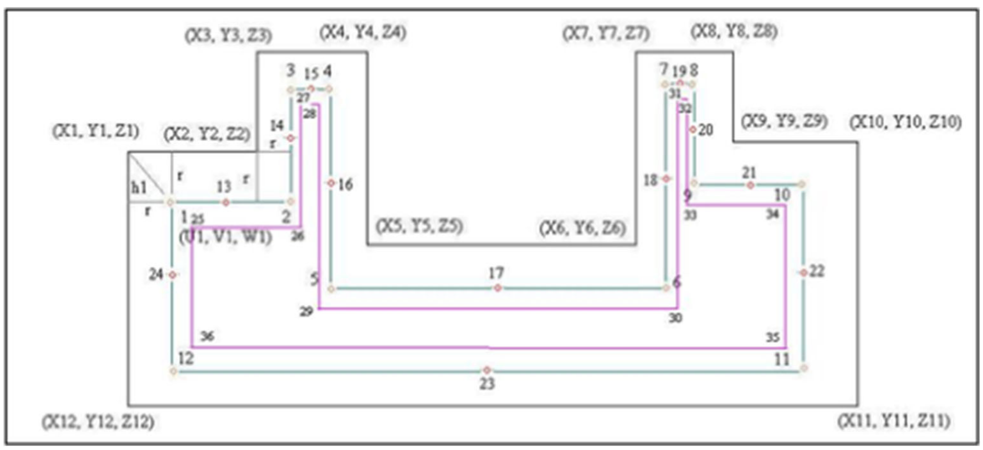
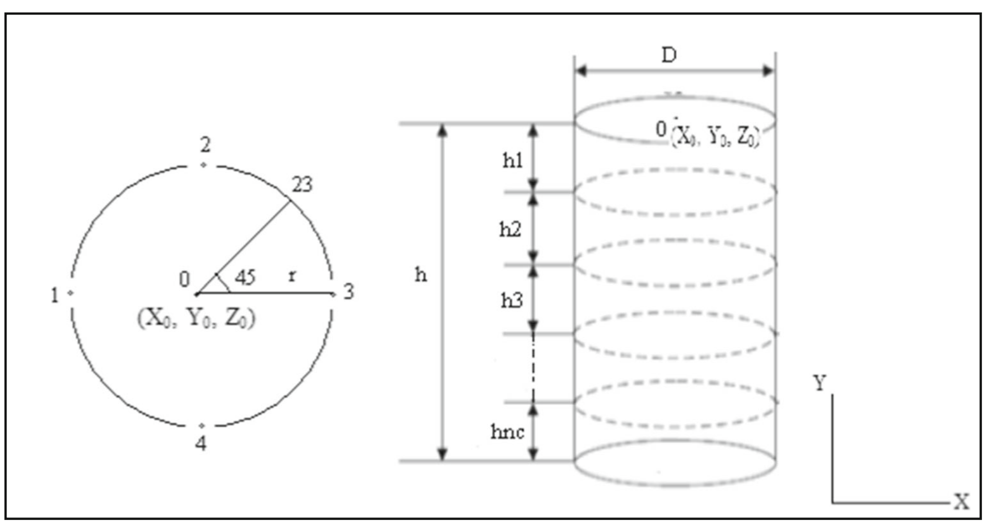


Fig. 26 Circular face for touch point generation



$$n_c = \sqrt{\frac{Nh}{2\pi r}} \text{ and } n_p = \frac{N}{n_c}$$

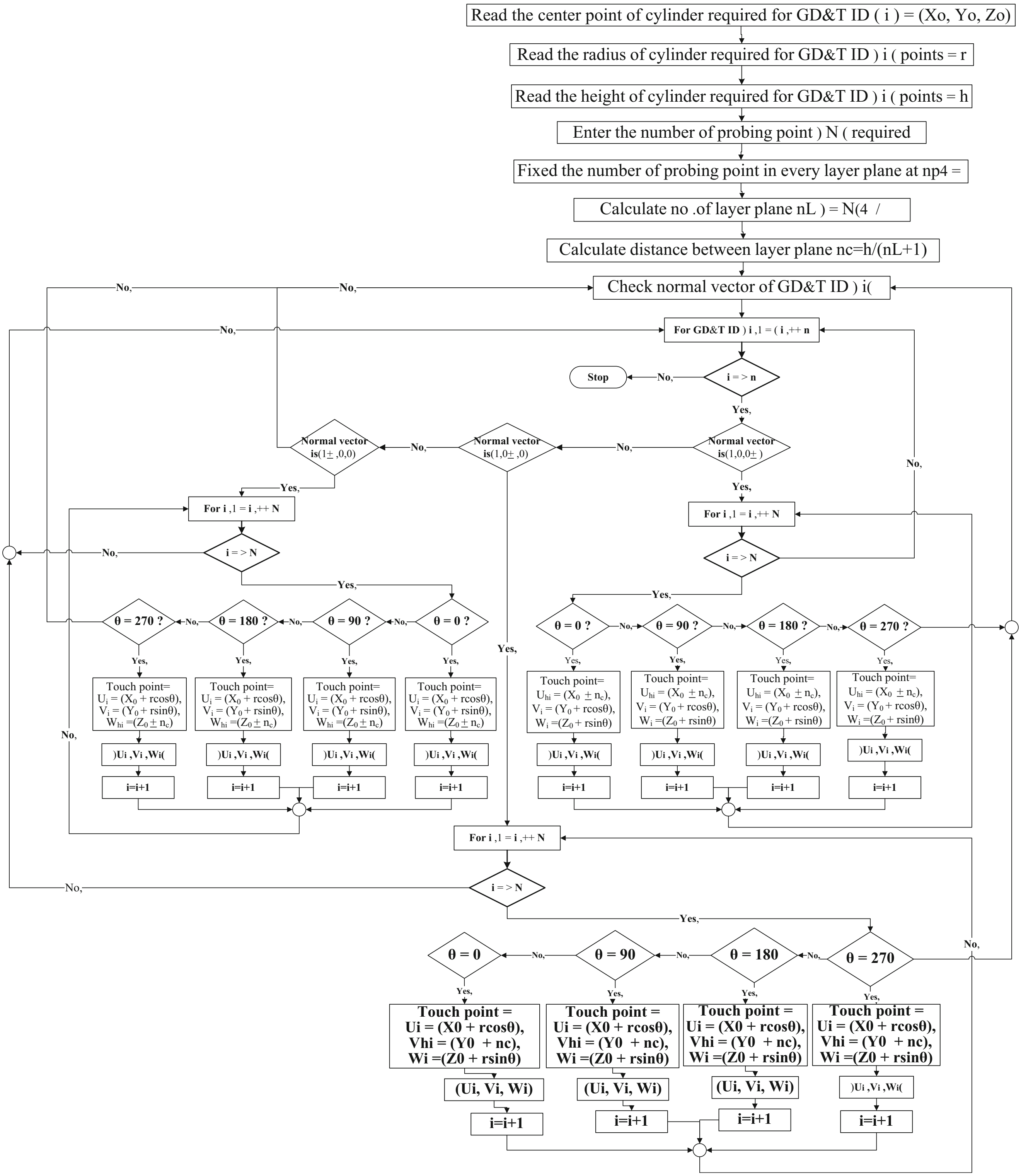


Fig. 27 Methodology for the generation of touch point on circular faces

Table 3 Table consisting of inspection plan

Face ID	ID of inspection operation	Tolerance	Tool used	Datum faces ID	Orientation of part	No. of touch point	Coordinates of touch point	Geometric inspection boundary
1	G&DT	Tolerance value	Probe type	Face ID	Setup no.	*n*	(*Ui*, *Vi*, *Wi*) – (*Ui*+1, *Vi*+1,*Wi*+1) – (*Ui*+2, *Vi*+2, *Vi*+2) – – (*Un*, *Vn*, *Vn*)	(*X*1, *Y*1, *Z*1) – (*X*2, *Y*2, *Z*2) – (*X*3, *Y*3, *Z*3) – (*X*4, *Y*4, *Z*4)

c. Finally, the probing point of the rectangular face will be as follows:

$$(U1,\ V1,\ W1);\ (U2,\ V2,\ W1);\ (U3,\ V3,\ W1);$$
$$(U4,\ V4,\ W1);\ (U5,\ V5,\ W1);$$
$$(U6,\ V6,\ W1);\ (U7, V7,\ W1);$$
$$(U8,\ V8,\ W1);\ (U9,\ V9,\ W1);$$
$$(U10,\ V10,\ W1);\ (U11,\ V11,\ W1);$$
$$(U12,\ V12,\ W1), (Un,\ Vn,\ W1)$$

Step 7 Similarly, coordinates for *YZ* and *ZX* planes can be computed to determine the probing points (*U*, *V*, *W*)

3.4.2 Touch point generation of circular face

Cylinder face as shown in Fig. 26 can be represented by using cylinder's center point (*X*0, *Y*0, *Z*0), its radius (*r*), and height (*h*). This information acts as the input to the algorithm which is explained as follows.

The procedure to determine the touch point for circular faces as shown in Fig. 27 can be explained as follows.

Step 1 Steps for the distribution of probing points

The n_c and n_p can be calculated as follows:

$$nc = \sqrt{\frac{Nh}{2\pi r}} \text{ and } n_p = \frac{N}{n_c}$$

Get the input variables from the extraction file such as:

N: number of pre-determined probing points on the cylinder surface
h: height of the cylinder
r: radius of the cylinder
n_c: Number of uniformly spaced layer planes perpendicular to the cylinder axis
n_p: Number of uniformly spaced measurements (probing points) at the intersection of the plane and the cylinder.

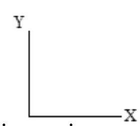
Step 2 Coordinates of the probing points

1. Input the location (X_0, Y_0, Z_0), radius, and height of the cylinder.
2. Dividing the cylinder height into n_c number of equally spaced layer planes:

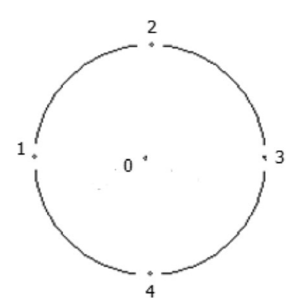
$$h_1 = h_2 = h_3 = h_4 = \ldots\ldots = h_{nc} = \frac{h}{n_c}$$

3. Now divide each layer plane into n_p number of equally spaced probing points.
4. In order to identify the plane on which the cylinder is lying, check the coordinates of the plane perpendicular to the axis of the cylinder. If all the *X*-coordinates are same, then the cylinder is intersecting with *YZ*-plane, if all the *Y*-coordinates are the same then the cylinder is on the *XZ*-plane and if all of the *Z*-coordinates are same, it means that the plane is lying on the *XY*-plane.
5. Assuming the cylinder is lying on the *XY*-plane with its axis in the *Z*-direction

a. For the first layer plane at h_1 height:



$Z_{h1}=Z_0-n_c$, considering only four probing points on the circumference of the layer plane and calculating their coordinates (X_1, Y_1, Z_{h1}), (X_2, Y_2, Z_{h1}), (X_3, Y_3, Z_{h1}), (X_4, Y_4, Z_{h1}) as follows:

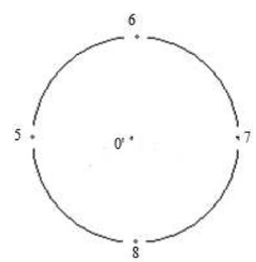


$$X_1 = (X_0 + r\cos\theta) \text{ and } Y_1 = (Y_0 + r\sin\theta); \text{at } \theta = 0$$
$$: X_1 = (X_0 + r) \text{ and } (Y_1 = Y_0)X_2$$
$$= (X_0 + r\cos\theta) \text{ and } Y_2 = (Y_0 + r\sin\theta); \text{at } \theta = 90$$
$$: (X_2 + X_0) \text{ and } Y_2 = (Y_0 + r)X_3$$
$$= (X_0 + r\cos\theta) \text{ and } Y_3 = (Y_0 + r\sin\theta); \text{at } \theta = 180$$
$$: X_3 = (X_0 - r) \text{ and } (Y_3 = Y_0)X_4$$
$$= (X_0 + r\cos\theta) \text{ and } Y_4 = (Y_0 + r\sin\theta); \text{at } \theta = 270$$
$$: (X_4 + X_0) \text{ and } Y_4 = (Y_0 - r)$$

Where: $Z_{h1}=Z_1=Z_2=Z_3=Z_4$, then, the probing point in the first layer plane: (X_1, Y_1, Z_{h1}); (X_2, Y_2, Z_{h1}); (X_3, Y_3, Z_{h1}); (X_4, Y_4, Z_{h1})

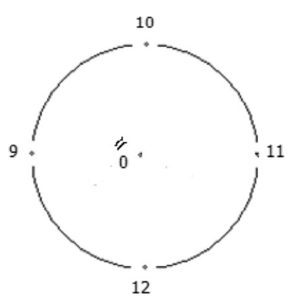
b. For the second layer plane at h_2 height:

$$Z_{h2} = Z_{h1} - n_c$$



c. For the third layer plane at h_3 height:

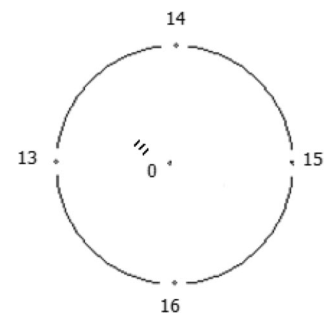
$$Z_{h3} = Z_{h2} - n_c$$



where: $(X_1=X_9, Y_1=Y_9)$; $(X_2=X_{10}, Y_2=Y_{10})$; $(X_3=X_{11}, Y_3=Y_{11})$; $(X_4=X_{12}, Y_4=Y_{12})$, and $Z_{h3}=Z_9=Z_{10}=Z_{11}=Z_{12}$, then, the probing point in the third layer plane: (X_9, Y_9, Z_{h3}); (X_{10}, Y_{10}, Z_{h3}); (X_{11}, Y_{11}, Z_{h3}); (X_{12}, Y_{12}, Z_{h3})

d. For the third layer plane at h_4 height:

$$Z_{h4} = Z_{h3} - n_c$$



Where: $(X_1=X_{13}, Y_1=Y_{13})$; $(X_2=X_{14}, Y_2=Y_{14})$; $(X_3=X_{15}, Y_3=Y_{15})$; $(X_4=X_{16}, Y_4=Y_{16})$, and $Z_{h4}=Z_{13}=Z_{14}=Z_{15}=Z_{16}$, then, the probing point in the third layer plane: (X_{13}, Y_{13}, Z_{h4}); (X_{14}, Y_{14}, Z_{h4}); (X_{15}, Y_{15}, Z_{h4}); (X_{16}, Y_{16}, Z_{h4})

e. Finally, the probing point of the cylinder at center point (X_0, Y_0, Z_0), radius (r), and height (h): (X_1, Y_1, Z_{h1}); (X_2, Y_2, Z_{h1}); (X_3, Y_3, Z_{h1}); (X_4, Y_4, Z_{h1}); (X_5, Y_5, Z_{h2}); (X_6, Y_6, Z_{h2}); (X_7, Y_7, Z_{h2}); (X_8, Y_8, Z_{h2}); (X_9, Y_9, Z_{h3}); (X_{10}, Y_{10}, Z_{h3}); (X_{11}, Y_{11}, Z_{h3}); (X_{12}, Y_{12}, Z_{h3}); (X_{13}, Y_{13}, Z_{h4}); (X_{14}, Y_{14}, Z_{h4}); (X_{15}, Y_{15}, Z_{h4}); and (X_{16}, Y_{16}, Z_{h4}).

   where: $(X_1=X_5, Y_1=Y_5)$; $(X_2=X_6, Y_2=Y_6)$; $(X_3=X_7, Y_3=Y_7)$; $(X_4=X_8, Y_4=Y_8)$, and $Z_{h2}=Z_5=Z_6=Z_7=Z_8$, then, the probing point in the second layer plane: (X_5, Y_5, Z_{h2}); (X_6, Y_6, Z_{h2}); (X_7, Y_7, Z_{h2}); (X_8, Y_8, Z_{h2})

f. Similarly, other points can be computed by assuming that the cylinder is lying in *YZ*-plane and *XZ*-plane.

3.5 Inspection planning table

Inspection planning table should contain all information obtained through feature extraction and recognition, accessibility analysis, setup planning, and determination of measuring points as shown in Table 3. This table can be explained as follows:

- Face ID is the ID number of the face in feature extraction and recognition output
- ID of the inspection operation is the type of GD&T like flatness, perpendicularity, etc.
- Tolerance value allows designers to set the tolerance limits for all of the various critical characteristics of the part by examine its function and its relationship to mating parts
- Tool used for inspection such as horizontal probe, vertical probe, star probe, etc., depending on the location, depth,

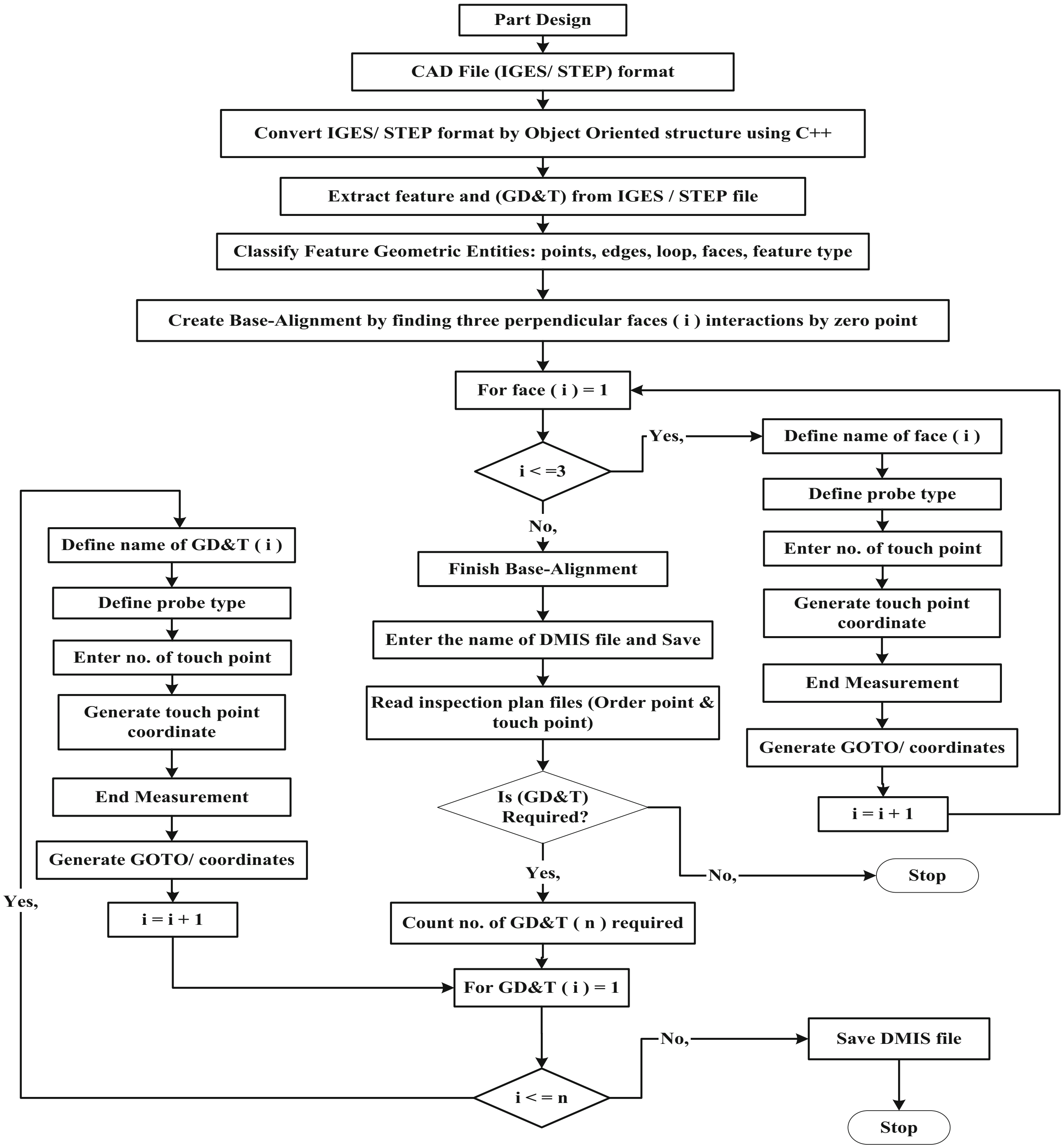


Fig. 28 Methodology for the generation of DMIS file

dimensions, and orientation of the features and radius, and length of the probe to avoid any collision between the part and the probe

- Datum faces are the reference points, lines, planes, and axis which are assumed to be exact
- Orientation of the part is the best positioning of the part and on the machine table
- Touch point are the coordinates of the probing point
- Geometric inspection boundary represents the boundary of the testing face

3.6 Inspection using coordinate-measuring machine (CMM)

Information in the inspection plan table cannot be used directly for measurement on CMM. It has to be converted to a

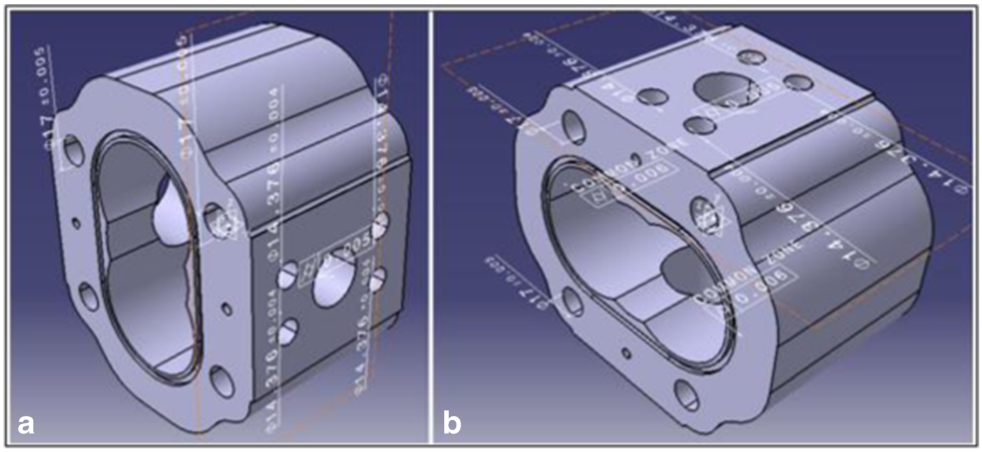


Fig. 29 Gear pump housing (**a**) first setup; (**b**) second setup

language that can be understood by the measurement machine. For this purpose, Dimensional Measuring Interface Standard (DMIS), a standard format of high-level programming language that has to be utilized, exists. It is used for bi-directional transfer of inspection data between CAD systems and the CMM. Methodology for the generation of DMIS file is presented in Fig. 28 and can be described as follows.

Table 4 Inspection plan for gear pump housing

Face ID	ID Inspection operation	Tolerance value	Tool used	Datum faces ID	Set up of the part	No. of touch points	Coordinates of touch points with directions	Geometric boundary of inspection face
FrHoleD	⌭	0.005	Star probe	----	SETUP1	8	(7,48.7,0,1,0,0); (0,48.7,7,0,0,1);	Center Point (0,77.4,0)
							(-7,48.7,0,-1,0,0); (0,48.7,-7,0,0,-1);	Raduis 7
							(7,51.9,0,1,0,0); (0,51.9,7,0,0,1);	Height 32
							(-7,51.9,0,-1,0,0); (0,51.9,-7,0,0,-1)	
ScHoleD	⌭	0.005	Star probe	----	SETUP1	8	(7,48.7,15.9,1,0,0); (0,48.7,22.9,0,0,1);	Center Point (0, 77.4, 15.9)
							(-7,48.7,15.9,-1,0,0); (0,48.7,8.9,0,0,-1);	Raduis 7
							(7,51.9,15.9,1,0,0); (0,51.9,22.9,0,0,1);	Height 32
							(-7,51.9,15.9,-1,0,0); (0,51.9,8.9,0,0,-1)	
ThHoleD	⌭	0.005	Star probe	----	SETUP1	8	(40.35,48.7,0,1,0,0); (33.35,48.7,7,0,0,1);	Center Point (0,77.4,15.9)
							(26.35,48.7,0,-1,0,0); (33.35,48.7,-7,0,0,-1);	Raduis 7
							(40.35,51.9,0,1,0,0); (33.35,51.9,7,0,0,1);	Height 32
							(26.35,51.9,0,-1,0,0); (26.35,51.9,0,-1,0,0)	
FuHoleD	⌭	0.005	Star probe	----	SETUP1	8	(- 40.35,48.7,15.9,1,0,0); (33.35,48.7,22.9,0,0,1);	Center Point (33.35,77.4, 15.9)
							(26.35,48.7,15.9,-1,0,0); (33.35,48.7,8.9,0,0,-1);	Radius 7
							(40.35,51.9,15.9,1,0,0); (33.35,51.9,22.9,0,0,1);	Height 32
							(26.35,51.9,15.9,-1,0,0); (33.35,51.9,8.9,0,0,-1)	
LfFHolD	⌭	0.005	Star probe	----	SETUP2	8	(45.16,8,45,0,1,0); (45.16,0,53,0,0,1);	Center Point (28.225,0,45)
							(45.16,-8,45,0,-1,0); (45.16,0,37,0,0,1);	Raduis 8
							(33.87,8,45,0,1,0); (33.87,0,53,0,0,1);	Height 112.9
							(33.87,-8,45,0,-1,0); (33.87,0,37,0,0,1)	
RgFHolD	⌭	0.005	Star probe	----	SETUP2	8	(-45.16,63,45,0,1,0); (-45.16,55,53,0,0,1);	Center Point (28.225,55.,45)
							(-45.16,47,45,0,-1,0); (-45.16,55,37,0,0,-1);	Raduis 15
							(-33.87,63,45,0,1,0); (-33.87,55,53,0,0,1);	Height 19.05
							(-33.87,47,45,0,-1,0); (-33.87,55,37,0,0,-1)	

Table 5 DMIS code for SETUP1

$$ CARL ZEISS - CALYPSO Preprocessor	F(ScHoleD_2)=FEAT/CYLNDR, INNER, CART, ,0,77.4,15.90,0,-1,0,14,32	F(FuHoleD_4)=FEAT/CYLNDR,INNER,CART, ,33.35,77.4,15.90,0,-1,0,14,32
$$ Ver.1.01.029.00 Date: Tue Apr 9 2013 Time: 13:55:58	MEAS / CYLNDR, F(ScHoleD_2), 8	MEAS / CYLNDR, F(FuHoleD_4), 8
DMISMN/'SSSSSSSSS'	PTMEAS/CART, 7,48.7,15.9,1,0,0	PTMEAS/CART, 40.35,48.7,15.9,1,0,0
FILNAM/'SSSSSSSSS.DMI'	PTMEAS/CART, 0,48.7,22.9,0,0,1	PTMEAS/CART, 33.35,48.7,22.9,0,0,1
UNITS/MM, ANGDEC, TEMPC	PTMEAS/CART, -7,48.7,15.9,-1,0,0	PTMEAS/CART, 26.35,48.7,15.9,-1,0,0
	PTMEAS/CART, 0,48.7,8.9,0,0,-1	PTMEAS/CART, 33.35,48.7,8.9,0,0,-1
SNSET/APPRCH, 2.0000	PTMEAS/CART, 7,51.9,15.9,1,0,0	PTMEAS/CART, 40.35,51.9,15.9,1,0,0
SNSET/RETRCT, 5	PTMEAS/CART, 0,51.9,22.9,0,0,1	PTMEAS/CART, 33.35,51.9,22.9,0,0,1
SNSET/SEARCH, 5.0000	PTMEAS/CART, -7,51.9,15.9,-1,0,0	PTMEAS/CART, 26.35,51.9,15.9,-1,0,0
	PTMEAS/CART, 0,51.9,8.9,0,0,-1	PTMEAS/CART, 33.35,51.9,8.9,0,0,-1
$$ Changing label 'Star~1' to: 'STAR1'.		
0.0000, 0.0000, $	GOTO/0,-5,77.4	GOTO/0,-5,77.4
-1.0000, 3.0050, SPHERE	GOTO/0,-5,20	GOTO/0,-5,20
	GOTO/0,-5,20	GOTO/5,-5,20
RECALL/DA(REFERENCE)	GOTO/0,-5,20	GOTO/5,28.225,20
MODE/PROG,MAN	GOTO/0,-5,77.4	GOTO/5,28.225,0
	ENDMES	ENDMES
F(FrHoleD_1)=FEAT/CYLNDR, INNER, CART, ,0,77.4,00,0,-1,0,14,32		
MEAS / CYLNDR, F(FrHoleD_1), 8	F(ThHoleD_3)=FEAT/CYLNDR, INNER, CART, ,33.35,77.4,00,0,-1,0,14,32	T(cylind_0)=TOL/CYLCTY, 0.005
PTMEAS/CART, 7,48.7,0,1,0,0	MEAS / CYLNDR, F(ThHoleD_3), 8	OUTPUT/FA(RightCy_0), TA(cylind_0)
PTMEAS/CART, 0,48.7,7,0,0,1	PTMEAS/CART, 40.35,48.7,0,1,0,0	
PTMEAS/CART, -7,48.7,0,-1,0,0	PTMEAS/CART, 33.35,48.7,7,0,0,1	T(cylind_1)=TOL/CYLCTY, 0.005
PTMEAS/CART, 0,48.7,-7,0,0,-1	PTMEAS/CART, 26.35,48.7,0,-1,0,0	OUTPUT/FA(FrHoleD_1), TA(cylind_1)
PTMEAS/CART, 7,51.9,0,1,0,0	PTMEAS/CART, 33.35,48.7,-7,0,0,-1	
PTMEAS/CART, 0,51.9,7,0,0,1	PTMEAS/CART, 40.35,51.9,0,1,0,0	T(cylind_2)=TOL/CYLCTY, 0.005
PTMEAS/CART, -7,51.9,0,-1,0,0	PTMEAS/CART, 33.35,51.9,7,0,0,1	OUTPUT/FA(ScHoleD_2), TA(cylind_2)
PTMEAS/CART, 0,51.9,-7,0,0,-1	PTMEAS/CART, 26.35,51.9,0,-1,0,0	
	PTMEAS/CART, 33.35,51.9,-7,0,0,-1	T(cylind_3)=TOL/CYLCTY, 0.005
GOTO/0,-5,77.4	GOTO/0,-5,77.4	OUTPUT/FA(ThHoleD_3), TA(cylind_3)
GOTO/0,-5,20	GOTO/0,-5,20	
GOTO/0,-5,20	GOTO/0,-5,20	T(cylind_4)=TOL/CYLCTY, 0.005
GOTO/0,-5,20	GOTO/0,-5,20	OUTPUT/FA(FuHoleD_4), TA(cylind_4)
GOTO/0,-5,77.4	GOTO/0,-5,77.4	
ENDMES	ENDMES	ENDFIL

Step 1. Read extraction file output and use it as input to generate the DMIS file automatically
Step 2. Generate base-alignment by finding zero point (base point of the part) which can be determined by using three interacting features on the part and the base-alignment will be closed
Step 3. Check if there GD&T are required
Step 4. If there GD&T are needed, total will be counted in the for loop as sum (N)
Step 5. Enter the name for DMIS file and the name for DMIS base-alignment file
Step 6. Enter number of touch points needed for the given GD&T (i)
Step 7. Read GD&T (i) face ID
Step 8. Read probe type ID depends on the probe accessibility analysis
Step 9. Read the probe path generation between selected features
Step 10. Read GD&T (i) type
Step 11. Read GD&T (i) tolerance value
Step 12. Read datum ID of the GD&T (i) and store it in the fifth column
Step 13. Make output of the test depending on the reference datum
Step 14. Finish the data required of the GD&T (i)
Step 15. Check if more GD&T are required or stop and finish DMIS file generation

Once the DMIS file is generated, it cannot be executed directly on CMM. It has to be post-processed for measuring the given part on the CMM.

3.7 Case study: gear pump housing

Application of the proposed techniques has been applied to measure the gear pump housing (mechanical pat) shown in Fig. 29a and b.

Inspection plan as shown in Table 4 and hence DMIS presented in Tables 5 and 6 has successfully been obtained for gear pump housing by implementing proposed methodologies and techniques. Finally, this DMIS programming code can be exported to CMM after post-processing for inspection.

Table 6 DMIS code for SETUP2

```
$$ CARL ZEISS - CALYPSO Preprocessor
$$ Ver.1.01.029.00 Date: Tue Apr 9 2013   Time: 13:59:58
DMISMN/'SAM'
FILNAM/'SAM.DMI'
UNITS/MM, ANGDEC, TEMPC
SNSET/APPRCH, 2.0000
SNSET/RETRCT, 5
SNSET/SEARCH, 5.0000

$$ Changing label 'Star~1' to: 'STAR1'.
S(STAR1)=SNSDEF/PROBE, INDEX, CART, -0.0150, 0.1975, -20.9113, 0.0000, 0.0000, $
 -1.0000, 3.0050, SPHERE
RECALL/DA(REFERENCE)
MODE/PROG,MAN
F(LfFHolD_5)=FEAT/CYLNDR,INNER,CART,28.225,0,45.0,1,0,0,16,-112.9
MEAS / CYLNDR, F(LfFHolD_5), 8
PTMEAS/CART, 45.16,8,45,0,1,0
PTMEAS/CART, 45.16,0,53,0,0,1
PTMEAS/CART, 45.16,-8,45,0,-1,0
PTMEAS/CART, 45.16,0,37,0,0,1
PTMEAS/CART, 33.87,8,45,0,1,0
PTMEAS/CART, 33.87,0,53,0,0,1
PTMEAS/CART, 33.87,-8,45,0,-1,0
PTMEAS/CART, 33.87,0,37,0,0,1
GOTO/5,28.225,0
GOTO/5,28.225,20
GOTO/-5,28.225,20
ENDMES
F(RgFHolD_6)=FEAT/CYLNDR,INNER,CART,,28.225,55.,45.0,-,0,0,16,112.9
PTMEAS/CART, -45.16,63,45,0,1,0
PTMEAS/CART, -45.16,55,53,0,0,1
PTMEAS/CART, -45.16,47,45,0,-1,0
PTMEAS/CART, -45.16,55,37,0,0,-1
PTMEAS/CART, -33.87,63,45,0,1,0
PTMEAS/CART, -33.87,55,53,0,0,1
PTMEAS/CART, -33.87,47,45,0,-1,0
PTMEAS/CART, -33.87,55,37,0,0,-1
ENDMES

T(cylind_5)=TOL/CYLCTY,  0.004
OUTPUT/FA(LfFHolD_5), TA(cylind_5)

T(cylind_6)=TOL/CYLCTY,  0.004
OUTPUT/FA(RgFHolD_6), TA(cylind_6)

ENDFIL
```

4 Conclusions

CMM has been a powerful tool in manufacturing industries due to its high inspection accuracy as well as its reduced inspection cost and time. This research has focused on various techniques and methodologies that can be used to generate an efficient inspection plan. Moreover, this work has resulted in following contributions:

- Present work has successfully integrated CAD, CAIP, and CMM systems through feature extraction and recognition, accessibility analysis, setup planning, measurement points, DMIS, etc. modules
- Feature extraction and recognition have successfully been carried out using both IGES and STEP file formats
- Methodologies based on PAD and ADD have been developed and verified successfully for accessibility analysis
- Two different techniques based on graphical method and ANN have been introduced and tested to perform setup planning, i.e., selection of best part orientation on CMM
- Simple and efficient algorithms have been proposed to determine measurement points on rectangular faces as well as circular faces
- DMIS and a postprocessor have been used to convert inspection plan table information into machine language for the purpose of CMM inspection
- A case study for successful measurement of gear pump housing has also been presented to show feasibility of proposed system and its techniques in real manufacturing applications
- Current application of proposed system has been limited to specific parts including prismatic parts and some of the axisymmetric parts. Therefore, its work can further be extended to include parts such as parts with inclined faces, complex axisymmetric parts, freeform-shaped parts, etc.

Acknowledgements This works is founded by National Plan for Science & Technology (NPST), Grant No. 10-INF1280-02, Saudi Arabia.

References

1. Limaiem. A, ElMaraghy. H.A, Automatic Planning for Coordinate Measuring Machines, Proceedings of IEEE International Symposium on Assembly and Task Planning Marina del Rey, CA - August 1997
2. Legge DI (1996) Integration of design and inspection systems—a literature review. Int J Prod Res 34(5):1221–1241
3. ElMaraghy HA, Gu PH, Bollinger JG (1987) Expert system for inspection planning. CIRP Ann Manuf Technol 36(1):85–89
4. Spitz. S.N, Spyridi. A.J, Requicha. A.A.G, Accessibility analysis for planning of dimensional inspection with coordinate measuring machines. IEEE Conference on Robotics and Automation, Los Angeles, USA, August 22–25, 1999.

5. Lee H, Cho M-W, Yoon G-S, Choi J-H (2004) A computer-aided inspection planning system for on-machine measurement—part I: global inspection planning. KSME Int J 18:1349–1357
6. Beg J, Shunmugam MS (2002) An object oriented planner for inspection of prismatic parts-OOPIPP. Int J Adv Manuf Technol 19: 905–916
7. Roy U, Xu Y, Wang L (1994) Development of an intelligent inspection planning system in an object oriented programming environment. Comput Integr Manuf Syst 7(4):240–246
8. Ghaleb YA, Ketan HS, Adil MB (2005) Integrating design and production planning with knowledge based inspection planning system. Arabia J Sci Eng 30(2B):245–260
9. Hwang CY, Tsai CY, Chang CA (2004) Efficient inspection planning for coordinate measuring machines. Int J Adv Manuf Technol 23: 732–742
10. Ziemian CW, Medeiros DJ (1998) Automating probe selection planning for inspection on measuring machine and part setup a coordinate. Int J Comput Integr Manuf 11(5):448–460
11. Fan KC, Leu MC (1998) Intelligent planning of CAD-directed inspection for coordinate measuring machines. Comput Integr Manuf Syst 11(1–2):43–51
12. Kweon S, Medeiros DJ (1998) Part orientations for CMM inspection using dimensioned visibility maps. Comput Aided Des 30(9):741–749
13. Corrigall MJ, Bell R (1991) Probe and component set-up planning for coordinate measuring machines. Int J Comput Integr Manuf 4(1):34–44
14. Cho MW, Lee H, Yoon GS, Choi JH (2004) A computer aided inspection planning system for on-machine measurement-part II: local inspection planning. KSME Int J 18(8):1349–1357
15. Zhang SG, Ajmal A, Wootton J, Chisholm A (2000) A feature-based inspection process planning system for co-ordinate measuring machine (CMM). J Mater Process Technol 107:111–118
16. Vafaeesefat A, EIMaraghy GA (2000) Automated accessibility analysis and measurement clustering for CMMs. Int J Prod Res 38(10): 2215–2231
17. Ajmal A, Zhang SG (1998) The application of a knowledge based clustering algorithm as an aid to probe selection and inspection process planning. Proc Inst Mech Eng: J Eng Manu 212:299–305
18. Chiang YM, Chen FL (1999) CMM probing accessibility in a single slot. Int J Adv Manuf Technol 15:261–267
19. Spyridi. A.J, Requicha. A.A.G, Accessibility analysis for the automatic inspection of mechanical parts by coordinate measuring machines, IEEE, 1999, LA
20. Spitz. S.N, Requicha. A.A.G, Multiple-Goals Path Planning for Coordinate Measuring Machines, Proceedings of IEEE International Conference on Robotics & Automation, April 2000
21. Lin YJ, Murugappan P (2000) A new algorithm for CAD-directed CMM dimensional inspection. Int J Adv Manuf Technol 16:107–112
22. Wu Y, Liu S, Zhang G (2004) Improvement of coordinate measuring machine probing accessibility. Precis Eng 28(1):89–94
23. Jiang BC, Chiu SD (2000) Form tolerance-based measurement points determination with CMM. J Intell Manuf 13:101–108
24. Devireddy CR, Ghosh K (1999) Feature-based modeling and neural networks-based CAPP for integrated manufacturing 1999. Int J Comput Integr Manuf 12(1):61–74
25. Aslan E, Seker U, Alpdemir N (1999) Data extraction from CAD model for rotational parts to be machined at turning centers. Turk J Eng Environ Sci 23(5):339–347
26. Wong FSY, Chuah KB, Venuvinod PK (2006) Automated inspection process planning: algorithmic inspection feature recognition, and inspection case representation for CBR. Robot Comput Integr Manuf 22:56–68
27. Lee JY, Kim K (1998) A feature-based approach to extracting machining features. Comput Aided Des 30(13):1019–1035
28. Nasr EA, Kamrani AK (2006) A new methodology for extracting manufacturing features from CAD system. Int J Comput Indust Eng l(51):389–415
29. Li WD, Ong SK, Nee AYC (2002) Recognizing manufacturing features from a design by feature model. Comput Aided Des 34: 849–868
30. Sunil VB, Agarwal R, Pande SS (2010) An approach to recognize interacting features from B-Rep CAD models of prismatic machined parts using a hybrid (graph and rule based) technique. Comput Ind 61:686–701
31. Tseng YJ (1999) A modular modeling approach by integrating feature recognition and feature-based design. Comput Ind 39:113–125
32. Fu MW, Ong SK, Lu WF, Lee IBH, Nee AYC (2003) An approach to identify design and manufacturing features from a data exchanged part model. Computer Aided Design l(35):979–993
33. Bespalov D, Regli WC, Shokoufandeh A (2006) Local feature extraction and matching partial objects. Comput Aided Des 38:1020–1037
34. Hunter. R, Guzman. M, Moller. J, Perez. J, A functional tolerance model: an approach to automate the inspection process, Journal of Achievements in Materials and Manufacturing Engineering 2008, 31 (2)
35. Nasr. E.A, Kamrani. A.K, Computer-based Design and Manufacturing: An Information-Based Approach, Published Book, Springer, 2007
36. Moroni G, Polini W (1998) Form feature selection in computer aided inspection planning. Int J flex Auto Integ Manu 6(1 & 2):29–51
37. Sathi. S.V.B, Rao. P.V.M, STEP to DMIS: Automated generation of inspection plans from CAD data. 5th Annual IEEE Conference on Automation Science and Engineering 2009, Bangalore, India, August 22–25