Robotics and Computer–Integrated Manufacturing 73 (2022) 102260

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

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

Full length article

Highly interacting machining feature recognition via small sample learning

Peizhi Shi, Qunfen Qi *, Yuchu Qin, Paul J. Scott, Xiangqian Jiang

EPSRC Future Advanced Metrology Hub, School of Computing and Engineering, University of Huddersfield, Huddersfield HD1 3DH, United Kingdom

ARTICLE INFO

Keywords:
Interacting feature recognition
Small sample learning
Single-shot refinement network
Deep learning

ABSTRACT

In the area of intelligent manufacturing, recognising the interacting features on a CAD model is a critical yet challenging task as topology structures of features are damaged due to the feature interaction. Some of the learning-based feature recognition methods produce less favourable results when recognising highly interacting features, while some require a significant amount of 3D models for training, which present an increasing challenge in a real world scenario, especially whenever collecting large training data becomes too difficult and time-consuming. To this end, effective highly interacting feature recognition via small sample learning becomes bottleneck for learning-based methods. To tackle the above issue, the paper proposes a novel method named RDetNet based on single-shot refinement object detection network (RefineDet) which is capable of recognising highly interacting features with small training samples. In addition, the paper further utilises several data augmentation (DA) strategies to increase the amount of relevant 3D training models. Experiments carried out in this paper show that the proposed method yields favourable results in recognising highly interacting features by using small training samples (e.g. 32 models per class).

1. Introduction

In Industrie 4.0, computer-aided technology plays an important part in automating manufacturing process [1]. The use of computer-aided technology to manufacture a component involves multiple stages, one of which is feature recognition. At this stage, machining features emerged in a computer-aided design (CAD) model are recognised based on its geometry information through algorithmic means.

Automatic feature recognition has been explored since 1980s. There are two methodologies in implementing such a system: rule-based [2] versus learning-based approaches [3]. To adopt the rule-based methods, researchers have to understand the topology structures of each machining feature and design ad-hoc rules for feature recognition accordingly. In the presence of interacting features, it is extremely challenging to understand the topology structures of an exhaustive list of feature combinations [3–5], which would lead to less robust and extendable rules [6]. Over the past decade, learning-based techniques and their applications gain growing popularity in the development of intelligent system (e.g. fault diagnosis [7], additive manufacturing [8,9], cloud manufacturing [10], object detection [11], 4D printing [12], human–robot collaboration [13,14]). Rather than manually designing heuristic rules, these techniques allow for learning rules from observable but less interpretable phenomenon (e.g. feature interaction). Therefore, a number of learning-based approaches (e.g. FeatureNet [15], MsvNet [16], SsdNet [17]) have been presented to solve the interacting feature recognition problem. However, some of these approaches produce less favourable results in recognising highly interacting features, while some of them require a huge amount of 3D models (e.g. 10 K models) for training. Those limitations present an increasing challenge to be used in a real-world application, especially when collecting large training data becomes too difficult and time-consuming. To this end, effective highly interacting feature recognition via small sample learning becomes a bottleneck for learning-based methods.

Motivated by the above limitations, this paper proposes a novel small sample learning approach named RDetNet based on a modified single-shot refinement object detection network (RefineDet) [18] for highly interacting feature recognition. On the one hand, this approach is able to discover meaningful information from small training samples. On the other hand, several data augmentation (DA) strategies are presented to further increase the diversity of 3D training models.

The main contribution of this paper is a small sample learning approach which is capable of yielding the favourable result in recognising highly interacting features by using small training samples.

The rest of paper is organised as follows. Section 2 makes an overview of existing interacting feature recognition approaches. Motivated by the research gaps, Section 3 proposes a small sample learning approach to interacting feature recognition. Section 4 makes a comparison between the proposed methods and others. Section 5 makes a conclusion for this paper.

* Corresponding author.
E-mail address: q.qi@hud.ac.uk (Q. Qi).

https://doi.org/10.1016/j.rcim.2021.102260
Received 11 March 2021; Received in revised form 12 September 2021; Accepted 14 September 2021
Available online 29 September 2021

2. Related work

In digital manufacturing, feature interaction refers to the situations where there are interactions between boundaries of machining features. Recognising interacting feature is a challenge task since 1980s as the interaction could result in changing the topology structures of machining features. To tackle this problem, researchers have presented a large number of approaches, which can be classified into rule-based and learning-based approaches [2,3,19]. This section gives an overview on these approaches to interacting feature recognition, and puts less weight on isolated feature recognition as this problem has been widely explored and solved during the past three decades.

2.1. Rule-based approaches

A typical rule-based method is called hint-based method [20]. In this approach, heuristic search is first carried out to extract the minimum indispensable fraction of each machining feature (i.e. hints) from a CAD model. An ad-hoc geometric reasoning and matching process is further employed to recognise the features based on the extracted hints. Since ad-hoc rules are adopted in the hint-based method, a major concern is that it is difficult to extend these rules to recognise new features. In the presence of highly interacting features, it is extremely challenging to define hints since the topology of these interacting features could be considerably damaged [6]. Therefore, multiple feature interpretations might be produced by the hint-based approaches [4].

Graph-based approach [21] is another feature recognition method. In this method, an important structure named attributed adjacency graph (AAG), which encodes the adjacency relationships of faces in a CAD model, is first created. A subgraph isomorphic matching procedure is further adopted to search all the potential features from the AAG. The original AAG suffers from several issues, such as inability to tackle curved faces [2] and intersecting features [4]. To support CAD models with curved surfaces, an advanced structure named multi-attributed adjacency graph (MAAG) that encodes more precise adjacency relations has been introduced by [22]. To tackle the interacting feature recognition issue, Gao and Shah [23] presented a hybrid approach that employs the graph- and hint-based ideas during feature recognition. This approach first divides the adjacency graph of a CAD model into a number of minimal condition subgraphs (MCSGs). Then, a hint-based geometric reasoning process is carried out to produce feature based on the given MCSG. However, only certain types of interacting features could be recognised by this approach [4]. In addition to the feature interaction issue, another common issue of the graph-based approaches is that the feature recognition process is time-consuming due to the adoption of heuristic search algorithm (i.e. subgraph matching) [4,6].

Another typical approach is called volumetric decomposition. In this approach, the removal volume of a CAD model is captured and decomposed into a number of intermediate volumes. Ad-hoc heuristic rules are further utilised to recognise the machining features according to these intermediate volumes. Based on the characteristics of the decomposition process, this type of approaches can be classified into convex hull decomposition [24] and cell-based decomposition approaches [25]. The former decomposes the volume into several convex volumes. The latter decomposes the volume into a number of cells, and merges those small cells into large volumes for further recognition. As multiple-step reasoning is adopted in this type of approaches, it is difficult to guarantee that each decomposition or recognition step is conducted correctly. Therefore, incorrect feature recognition results might be produced [4].

In addition to the aforementioned approaches, machining features can also be recognised via other rule-based methods (e.g. planning [26], STEP-based [27–29], syntactic pattern recognition [30], ontology-based [31,32] approaches). A distinguishing characteristic of these approaches is the use of heuristic rules for reasoning, searching, modelling, or planning during feature recognition. Thorough knowledge about feature interaction is essential for the success of feature recognition. Therefore, these methods normally face the similar issues, i.e. inflexibility, and difficulty in tackling highly interacting features [17].

2.2. Learning-based approaches

As discussed in the previous section, comprehensive knowledge about feature interaction is demanded in the rule-based methods in order to implement reliable rules. As feature interaction is observable but less interpretable, it should be easier to learn rules via machine learning techniques from the CAD models than manually design ad-hoc rules [3]. Therefore, several learning-based methods have been presented to tackle the interacting feature recognition problem.

A typical learning-based interacting feature recognition method is the FeatureNet [15]. This approach first adopts watershed algorithm to extract features from a voxelized CAD model, and then utilises a pre-trained 3D convolutional neural network (CNN) to recognise the extracted features. The FeatureNet could be classified as a two-stage method where feature extraction and recognition are not carried out simultaneously [17]. Due to the nature of watershed algorithm employed at the feature extraction stage, this approach is able to produce satisfactory results when recognising features with low interaction degree, but may mis-extract and mis-recognise some highly interacting features since the topology structures of highly interacting features are damaged [17]. In addition to the FeatureNet, another learning-based two-stage method [6] has been proposed recently. This approach first adopts a heuristic hint-based feature extractor to produce potential features based on the given CAD model. Then, a graph learning framework taking a 2D CNN structure is used for feature recognition. As the proposed network was trained by using simple CAD models with 15 pairs of interacting features, it is unclear whether this approach is applicable to models with multiple highly interacting features. Other two-stage methods (e.g. [16,33,34]) also suffer from a similar problem, i.e. difficulty in tackling different types/degrees of feature interactions.

To solve the above problem, Shi et al. proposed a one-stage method named SsdNet [17], in which feature extraction and recognition are carried out simultaneously via supervised learning algorithm. In this approach, six viewing images are collected from the six viewing directions of a given CAD model, and then passed through a 2D CNN for feature extraction and recognition. The machining features captured from six viewing directions are integrated together to yield a final result. This method is able to recognise highly interacting features with ease in spite of the fact that their topology structures are considerably damaged. In addition, the SsdNet only needs the 3D models with isolated features rather than the models with interacting features for training, which can also be considered as an advantage. One major concern is that the number of required 3D isolated feature models (i.e. 512 models per class, 12,288 models in total) at the training stage is so large that it is difficult to collect in a real world application.

3. Methodology

As aforementioned, effective highly interacting feature recognition via small sample learning becomes a bottleneck of learning-based methods. This section presents an advanced methodology to address the above problem. This paper assumes that limited number of voxelized isolated feature models are available for training as models with isolated machining feature are easy to be collected in practice. 24 machining features adopted in [15] are selected to validate the proposed method.

The experiments conducted in the SsdNet [17] demonstrate that a one-stage view-based method is capable of yielding favourable results in recognising highly interacting features. This network directly maps a viewing image collected from one viewing direction of a 3D interacting feature model to a set of features in the viewing direction. The final feature recognition result is yielded by merging features detected from six directions together. Motivated by such an observation, view-based idea is also adopted in this paper (see the interface phase in Fig. 1). To enable the small sample learning, a potential direction is to employ advanced machine learning technique to discover meaningful information

Table 1
DA strategies.

	DA operation description.
1	Mirror the 3D model along the front plane.
2	Mirror the 3D model along the right plane.
3	Push the feature downward.
4	Pull the feature upward.
5	Move the feature leftward.
6	Move the feature rightward.
7	Move the feature frontward.
8	Move the feature backward.
9	Apply edge padding to the front-left corner of the 3D model.
10	Apply edge padding to the front-right corner of the 3D model.
11	Apply edge padding to the back-left corner of the 3D model.
12	Apply edge padding to the back-right corner of the 3D model.
13	Crop the front-left corner of the 3D model.
14	Crop the front-right corner of the 3D model.
15	Crop the back-left corner of the 3D model.
16	Crop the back-right corner of the 3D model.

from small training samples. Another straightforward approach is data augmentation (DA). This technique is capable of increasing the amount of 3D training models, and has been proven effective in the feature recognition task [16]. To this end, the proposed framework consists of two main components: DA approach to constructing the training set, and view-based small sample learning network (see the training phase in Fig. 1). The following sections illustrate these two components respectively.

3.1. Training set construction

As noted, this approach employs the view-based supervised learning technique for feature extraction and recognition. Therefore, this section illustrates how to create a diverse training dataset which contains viewing images of interacting feature models and their corresponding labels.

An assumption made in this paper is that limited number of voxelized isolated feature models are available for training. Each 3D model in this set consists of only one feature. The proposed method, however, aims to recognise interacting features. Therefore, this paper synthesises interacting feature models from the isolated feature models, and to capture 2D viewing images from those synthetic interacting feature models. Fig. 2 (A) illustrates the dataset construction process. This approach first selects 2–10 voxelized isolated feature models from the dataset randomly. The selected models are further augmented by a number of DA operations to increase the model diversity, which will be discussed later on, and incorporated into a model with interacting features via a logical operation. Six viewing images and their corresponding labels are then collected from six viewing directions of the interacting feature model, as suggested in [17].

As stated in Section 3, DA technique is capable of increasing the amount of the 3D isolated feature models, and has been proven effective in the feature recognition task [16]. However, the DA techniques adopted in [16] suffer from several issues, e.g. lack of robustness, inability to produce highly diverse models. Motivated by work in [16], this paper presents a number of DA operations as shown in Table 1 and Fig. 3. Like the DA techniques adopted in [16], the proposed DA operations may also yield corrupted features (e.g. a half triangular passage). To avoid this problem, a simple algorithm, which discards those corrupted features if six faces of the 3D model are damaged by the DA operations, is implemented. In addition, some operations (e.g. push, pull) are disabled for specific feature class (e.g. through hole) since these operations are not applicable on that feature class, as shown in Table 2. The above techniques allow for producing rich and diverse 3D models without introducing corrupted features. The benefits of the proposed DA techniques will be examined in Section 4.3.

Table 2
DA strategies utilised for different feature classes.

	Feature name	DA strategies				
		Mirror	Push/pull	Move	Padding	Crop
1	O ring	✓	✓	✓	✓	✓
2	Through hole	✓		✓	✓	✓
3	Blind hole	✓	✓	✓	✓	✓
4	Triangular passage	✓		✓	✓	✓
5	Rectangular passage	✓		✓	✓	✓
6	Circular through slot	✓	✓	✓	✓	✓
7	Triangular through slot	✓	✓	✓	✓	✓
8	Rectangular through slot	✓	✓	✓	✓	✓
9	Rectangular blind slot	✓	✓	✓	✓	✓
10	Triangular pocket	✓	✓	✓	✓	✓
11	Rectangular pocket	✓	✓	✓	✓	✓
12	Circular end pocket	✓	✓	✓	✓	✓
13	Triangular blind step	✓	✓			✓
14	Circular blind step	✓	✓			✓
15	Rectangular blind step	✓	✓	✓	✓	✓
16	Rectangular through step	✓	✓	✓	✓	✓
17	2-sides through step	✓	✓			✓
18	Slanted through step	✓	✓			✓
19	Chamfer	✓	✓	✓	✓	✓
20	Round	✓	✓	✓	✓	✓
21	Vertical circular end blind slot	✓	✓	✓	✓	✓
22	Horizontal circular end blind slot	✓	✓			✓
23	6-sides passage	✓		✓	✓	✓
24	6-sides pocket	✓	✓	✓	✓	✓

3.2. Small sample learning network

This paper aims to implement a one-stage view-based small sample learning network. The proposed neural network takes a 2D image collected from one viewing direction of a 3D interacting feature model as input, and outputs a set of 3D features in the viewing direction. To implement this idea, the paper adopts single-shot refinement object detection network (RefineDet) [18] as it is capable of obtaining meaningful information from small training samples effectively. Some adjustments have been applied to the network architecture to fulfil the view-based idea. This section presents the modified network architecture, learning and feature recognition processes in details.

3.2.1. Network architecture

Before showing details of the network, this section first discusses a key concept in the area of object detection, i.e. *anchor box* (or prior box) [35]. An anchor box refers to a predefined bounding box which represents the ideal location, shape and size of an object emerged in an image, as shown in Fig. 4 (A). Instead of predicting the actual locations, shapes and sizes of objects, an anchor-based object detection algorithm (e.g. YOLO [36], SSD [35]) normally constructs thousands of anchor boxes, and predicts relative offsets between each pre-defined anchor box and its corresponding object. Despite the high computational efficiency, anchor-based idea faces a number of issues (e.g. huge search space) which could affect the recognition performances [18].

RefineDet [18] is an effective object detection technique which can solve issues arising from the anchor-based approaches. The original RefineDet maps an image with a dimension (3, 320, 320) to a set of 2D boxes. In this research problem, however, the neural network is supposed to predict 3D feature boxes. Therefore, this paper modified the ARM and ODM of the network to allow for outputting 3D feature boxes based on 2D viewing images. In addition, the network architecture of the original RefineDet is finely adjusted in order to speed up the training and testing process. To this end, the modified network named RDetNet is capable of mapping a viewing image of dimension (3, 64, 64) to 4080 3D boxes, as shown in Fig. 5. The benefits of such an adjustment will be examined in Section 4.3.

It is observed that the network involves two modules. The first module is called anchor refinement module (ARM). This module adopts a viewing image as input, and coarsely predicts the locations and

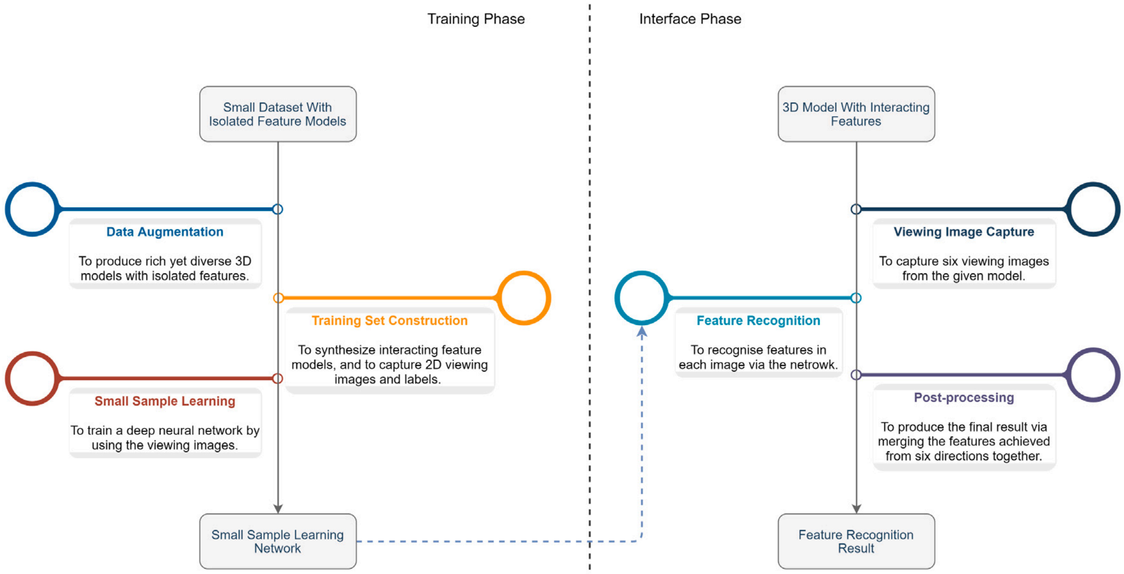


Fig. 1. The system diagram of the proposed method.

existences of features emerged in the given viewing image. The output dimension of ARM is $(5+2) \times 4080$, where 4080 is the number of pre-defined anchor boxes, 5 is offset values between a machining feature and the anchor box (i.e. height, width, depth, and centre coordinate offsets), 2 is a binary confidence score which indicates whether this box contains a machining feature. Several negative anchors (i.e. anchors without machining features) are filtered out at this stage, while these predicted coarse locations of machining features, defined as *refined anchor boxes*, will be utilised in the subsequent module (see Fig. 4 (B)). On the one hand, this module is capable of reducing the search space for feature recognition as a large number of negative anchors haven been eliminated at this stage. On the other hand, these refined anchors produce good initialisation for the subsequent module. The second module involved in the network is called object detection module (ODM), which is designed to get accurate class and location of each machining feature emerged in the viewing image according to the refined anchors generated in the previous module. The output dimension of ODM is $(5+25) \times 4080$, where 4080 is the number of refined anchor boxes achieved from the ARM, 5 is offset values between a machining feature and the refined anchor box, 25 is the number of feature classes, i.e. 24 machining feature classes and a background class (see Fig. 4 (C)).

The ARM and ODM are linked together via a number of transfer connection blocks (TCBs). These blocks can transfer the activation maps achieved from ARM to ODM (see Fig. 6 for details). As suggested in [18], a deconvolution operation is adopted to increase the dimension of the activation maps achieved from higher layers, and the element-wise operation is employed to sum the activation maps from different layers. This paper adjusts the parameters of the deconvolution layers in the original RefineDet [18] in order to speed up the training process.

3.2.2. Training

An essential step during the training is to establish the connection between an anchor box and a machining feature. To tackle this problem, the paper utilises intersection over union metric to evaluate the overlap between an anchor box and a machining feature. This approach matches the machining feature to an anchor with the highest overlap value, and matches each of the rest anchors to a machining feature when the overlap value between the anchor and feature is larger than 0.5. The remaining anchor boxes are categorised into negative anchors. As suggested in [18], hard negative mining is adopted in this paper to select negative anchors with high loss scores to guarantee the ratio between positive anchors and negative anchors is around 1:3. When training the ODM, a negative refined anchor box is discarded if the confidence score of this box achieved from the ARM is lower than a pre-defined threshold.

The training loss for all matched anchors is as follows:

$$L = \frac{1}{N_1}\Big(\sum_i L_b(p_i, [l_i^* > 0]) + \sum_i [l_i^* > 0] L_r(x_i, g_i^*)\Big) + \frac{1}{N_2}\Big(\sum_i L_m(c_i, l_i^*) + \sum_i [l_i^* > 0] L_r(t_i, g_i^*)\Big) \quad (1)$$

where N_1 and N_2 refer to the number of matched anchors in ARM and ODM, i refers to the matched anchor index, g_i^* refers to the actual feature location offsets relative to the anchor (or refined anchor) i, l_i^* refers to the class of anchor (or refined anchor) i, p_i and x_i refer to the predicted binary confidence scores and location offsets of refined anchor in ARM, c_i and t_i refer to the predicted confidence scores and location offsets of the machining feature in ODM. L_b, L_m and L_r refer to the binary cross-entropy loss, softmax loss and $L1$ regression loss. $[l_i^* > 0]$ equals to one if the anchor is positive, and equals to zero otherwise.

As suggested in [17], this approach also utilises a neural network trained on the pascal set [37] for parameter (i.e. weights and biases in the RDetNet) initialisation, and employs three image operations (i.e. random flipping, resizing, and combination) for online data augmentation. During the training, Adam [38] is used to optimise the loss function as this optimiser is capable of producing good learning results.

3.2.3. Feature recognition

At the feature recognition stage, six viewing images are first captured from the given 3D CAD model, and passed through the RDetNet. For each viewing image, the ARM eliminates several negative anchor boxes, and produces a number of refined anchor boxes. The ODM then adopts the refined anchors to output 4080 3D bounding boxes per viewing image. Finally, the algorithm merges the bounding boxes achieved from six directions together, and eliminates redundant machining features via soft non-maximum suppression [39], as carried out in [17]. The feature recognition process is shown in Fig. 7.

4. Experimental results

This section reports the experimental results on the interacting machining feature recognition. An exhaustive evaluation was carried out to compare RDetNet with others, and to demonstrate the benefits of the presented learning strategies.

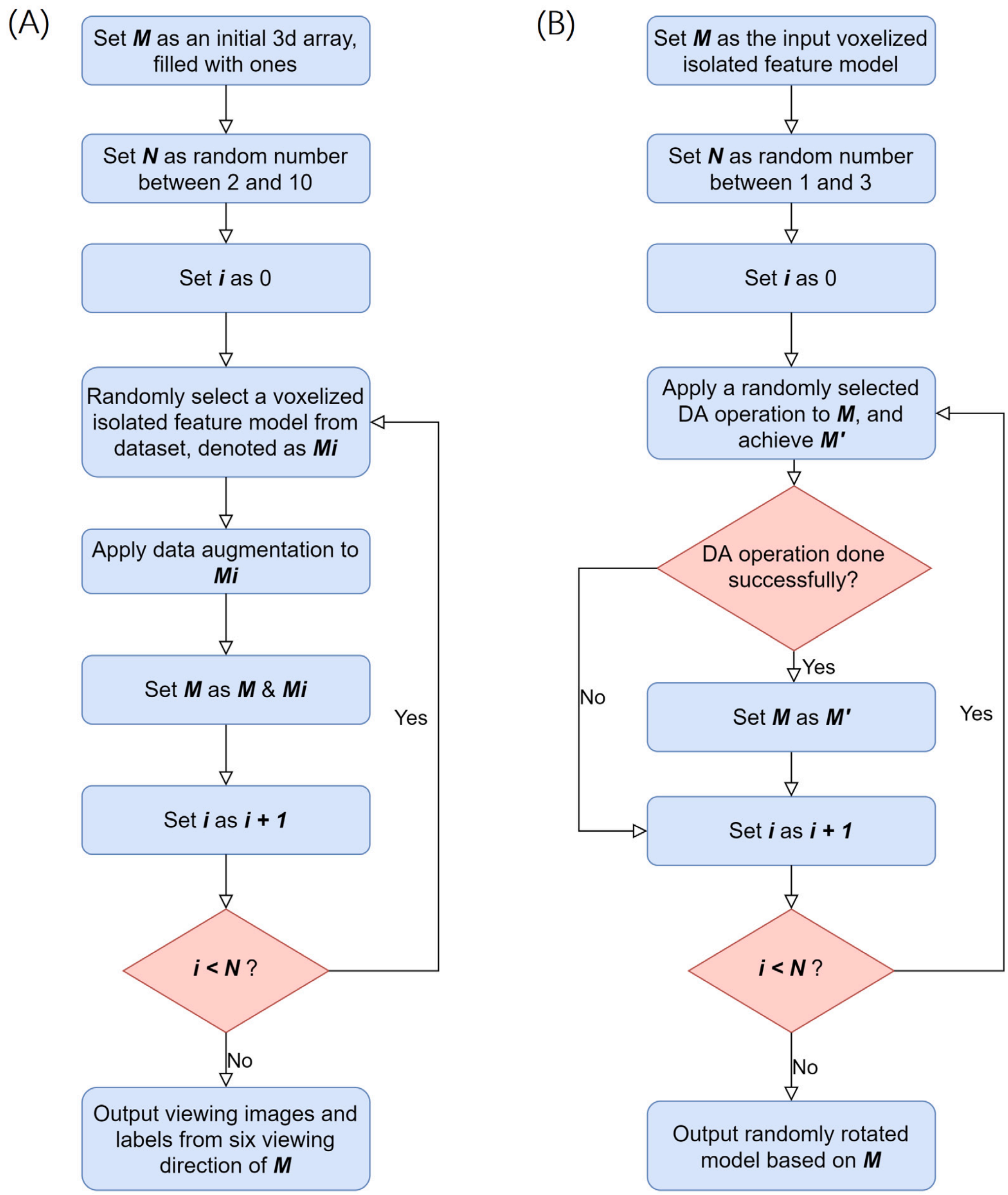


Fig. 2. The system flowcharts of (A) interacting feature model construction and (B) data augmentation.

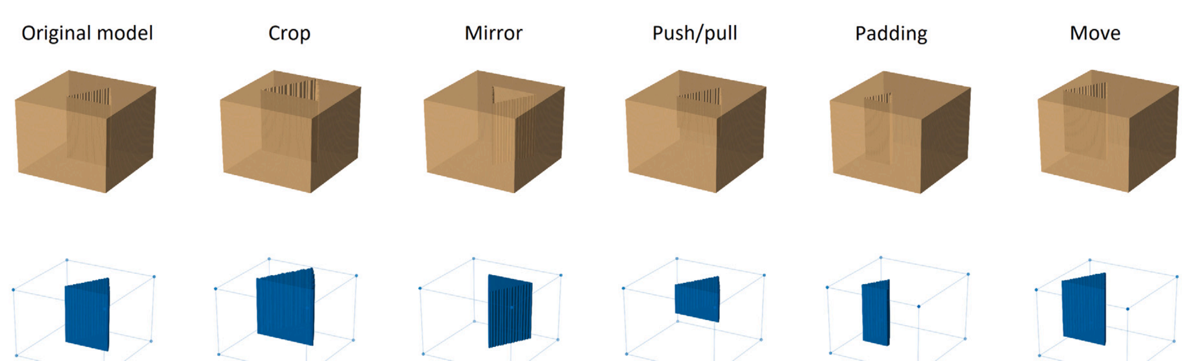


Fig. 3. Data augmentation strategies.

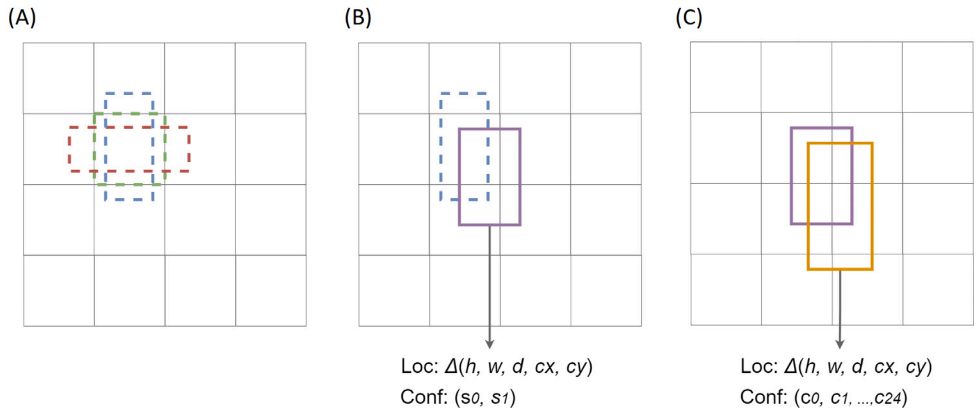


Fig. 4. Anchors and machining feature boxes. (A) An activation map and pre-defined anchors (boxes with dotted line). (B) Refined anchor box produced by ARM (box with purple line) and its corresponding pre-defined anchor (box with dotted line). $\Delta(h, w, d, cx, cy)$ are offset values between a refined anchor and the pre-defined anchor box (i.e. height, width, depth, and centre coordinate offsets), (s_0, s_1) are binary confidence scores which indicate whether this refined anchor box contains a machining feature. (C) Refined anchor box produced by ARM (box with purple line) and feature box produced by ODM (box with yellow line). $\Delta(h, w, d, cx, cy)$ are offset values between a machining feature and the refined anchor box, $(c_0, c_1, \ldots, c_{24})$ are the feature class confidence scores.

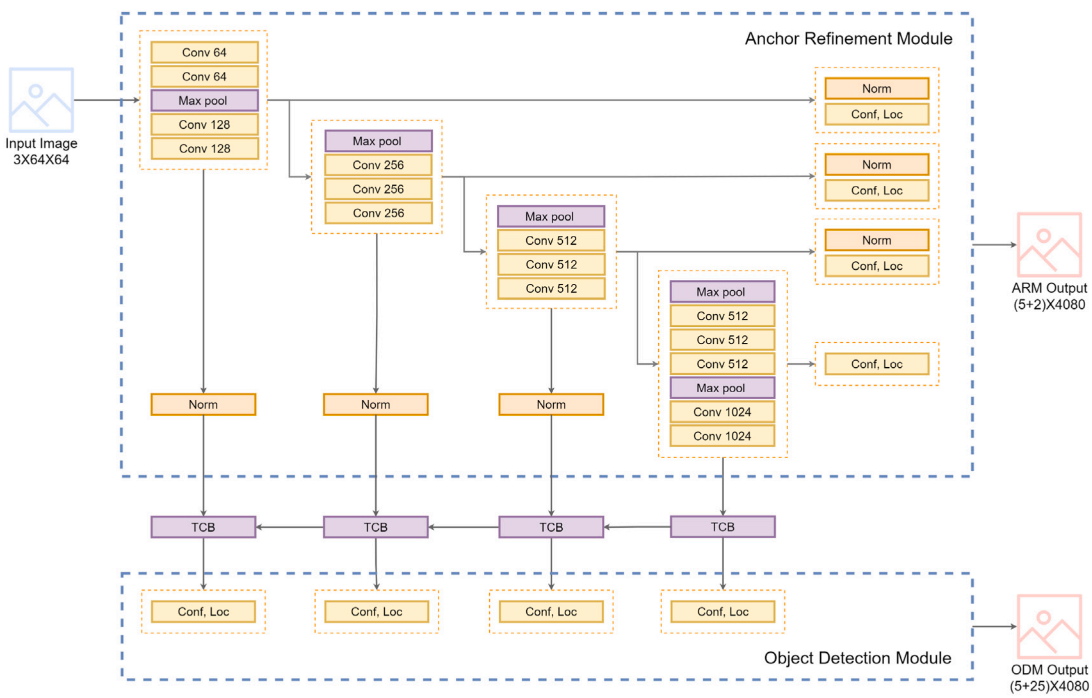


Fig. 5. RDetNet architecture.

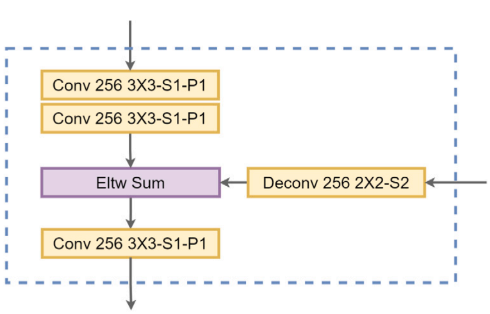


Fig. 6. Transfer connection block [18].

4.1. Experimental configurations

The study conducted in this section compares the proposed method with a number of learning-based approaches (i.e. FeatureNet [15], MsvNet [16], and SsdNet [17]). The FeatureNet [15] and MsvNet [16] are two-stage methods, while SsdNet [17] is the state-of-the-art learning-based one-stage approach which was published recently. These sophisticated learning-based approaches are chosen since they are general purpose approaches which are able to tackle different kinds of feature interactions. In addition, these approaches can be trained/tested using the same dataset with STL models, which allows for a fair comparison. As discussed in Section 3 and [15–17], both the RDetNet and other methods (i.e. SsdNet, MsvNet and FeatureNet) adopt a dataset of isolated machining features for training/validation purpose, and utilise a dataset of interacting machining features for testing. In this paper, the *isolated feature set* [15] and *interacting feature set* [16] are utilised to conduct the evaluation since these two public datasets, which are

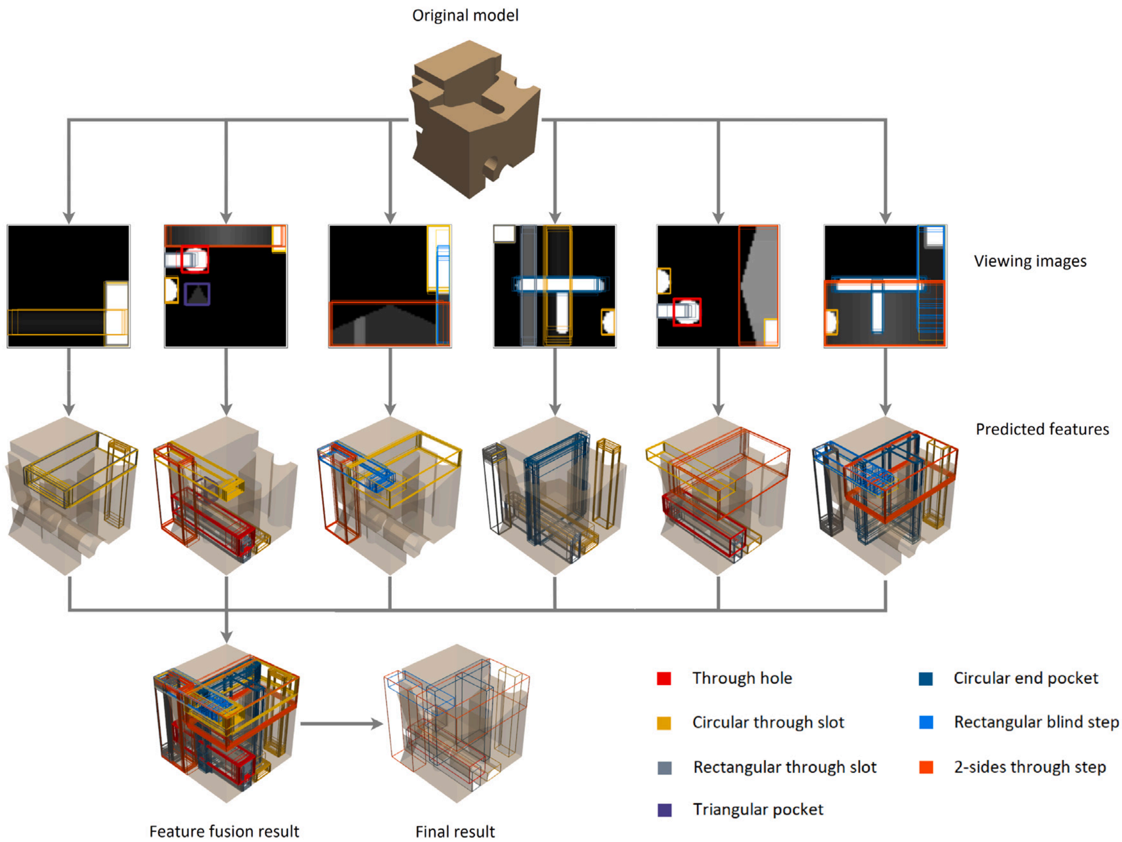


Fig. 7. Feature recognition process.

adopted as benchmarks in [16,17], contain rich yet diverse classes of machining features (24 classes). The former set is composed of 24 K 3D STL models with isolated machining features (1 K models per class), which are utilised at the training and validation stages. The latter set is composed of 1 K 3D STL models with highly interacting machining features, which are employed for testing and comparison purpose. All the STL models in the two sets are exemplified in Fig. 8. As suggested in [15–17], this paper voxelizes these models into arrays with dimensions (64, 64, 64) via a library named *binvox*.

90% models in the isolated feature set are randomly selected to create a training set, while the rest models in the isolated feature set are utilised to create a validation set. The training and validation sets involve 900 and 100 isolated feature models in each class respectively. The whole interacting feature set is utilised as a test set, which involves 1000 models with interacting features. To test the recognition accuracies of above approaches yielded by using different numbers of training models, not all 900 models in the training set are selected at the training stage. Instead, only N_{train} models per class from the training set are utilised for training (e.g. $N_{train} = 32$). The detailed information about N_{train} in each experiment is reported in the following sections. The data configurations adopted in this paper are identical to those suggested in [16,17] in order to make a fair comparison.

The RDetNet adopts 2D greyscale images rather than 3D voxel models as inputs of deep neural network. To train this network, a set with 320 K 2D greyscale images and a set with 1.6 K images are constructed for training and validation purposes via the process presented in Section 3.1. This paper sets the batch size, number of learning epochs and learning rate as 16, 6 and 10^{-4}. A learning rate decay strategy is utilised to drop the rate by the factor of 10 at the 4th epoch as suggested in [17,40]. Above hyper-parameters are turned based on the validation performances. The optimal configurations are utilised in the SsdNet, MsvNet and FeatureNet. A RTX 2080ti graphics card is employed to speed up the training and testing processes.

By following the suggestions in [17], F-score is selected to measure the recognition accuracy of different approaches as this metric is capable of providing deep insights into different machining feature detectors. The F-score (denoted as $\mathcal{F}$) is defined as follows:

$$\mathcal{F} = \frac{2\mathcal{P}\mathcal{R}}{\mathcal{P}+\mathcal{R}}, \tag{2}$$

$$\mathcal{P} = \frac{1}{K}\sum_{i=1}^{K}\frac{\sum_{j=1}^{N} tp_{i,j}}{\sum_{j=1}^{N} pred_{i,j}}, \tag{3}$$

$$\mathcal{R} = \frac{1}{K}\sum_{i=1}^{K}\frac{\sum_{j=1}^{N} tp_{i,j}}{\sum_{j=1}^{N} gt_{i,j}}, \tag{4}$$

$$tp_{i,j} = \min(pred_{i,j}, gt_{i,j}) \tag{5}$$

where $\mathcal{P}$ and $\mathcal{R}$ refer to the precision and recall scores, N and K refer to the number of test models and feature classes, $tp_{i,j}$ refers to the amount of accurately predicted class i features emerged in model j (true positive value), $pred_{i,j}$ refers to the amount of predicted class i feature emerged in the model j (predicted value), and $gt_{i,j}$ is the real amount of the class i feature emerged in the model j (ground truth). In addition to the F-score, the average time in recognising a CAD model with interacting features is also employed as the evaluation metric since it could evaluate the real-time recognition performances of different systems.

4.2. Comparative study

This section makes a comparison between the proposed approach and others. F-score and the average recognition time are adopted as evaluation metrics.

To achieve the recognition results of the RDetNet and SsdNet yielded by using different amount of training models, N_{train} 3D models per class are randomly selected from the isolated feature dataset

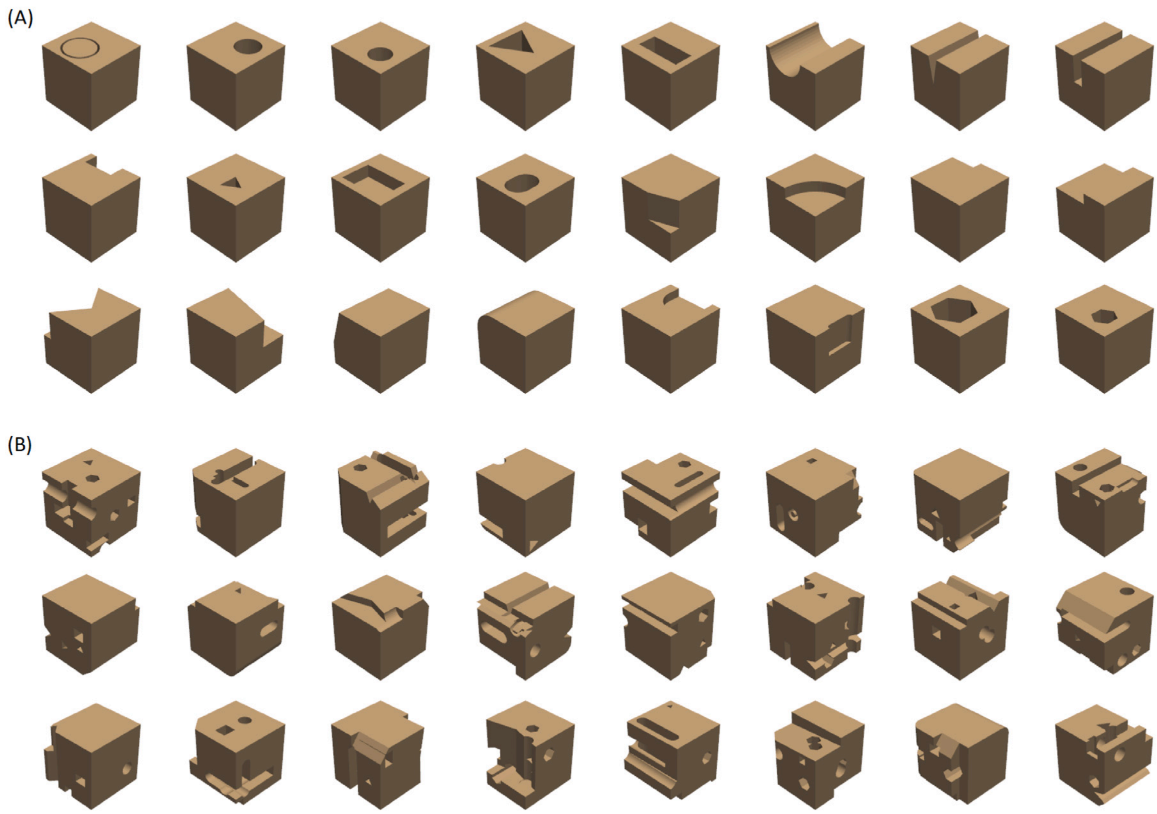


Fig. 8. Datasets utilised in this experiment: (A) isolated feature set [15], and (B) interacting feature set [16].

Table 3
The average recognition time (milliseconds per model) taken by four approaches.

	RDetNet	SsdNet	MsvNet	FeatureNet
Average time	138.71	111.98	548.30	329.77

for training ($N_{train} \in \{2^5, 2^6, 2^7, 2^8, 2^9\}$ in this experiment). As two benchmarks, the FeatureNet and MsvNet, however, are trained under the default optimal settings suggested in [15,16] where 512 (= 2^9) 3D models per class are utilised for training.

Fig. 9 illustrates the recognition F-scores yielded by the four methods. From this figure, it is observed that the RDetNet outperforms the SsdNet by utilising any amount of isolated feature models for training. It is also evident that the proposed approach could yield near-optimal result (i.e. $\mathcal{F} = 93.17\%$) by using only a few 3D isolated feature models (i.e. $2^5 = 32$ models per class) for training. The result implies that the RDetNet only requires 32 STL models with isolated machining features per class in a real-world industrial scenario, which might not be difficult to collect. In addition, both the proposed approach and the SsdNet outperform two benchmark approaches (the MsvNet [16] and FeatureNet [15]) under the optimal settings in terms of recognition F-scores. The feature recognition results yielded by the RDetNet trained with 32 models per class are illustrated in Fig. 10. It is observed that the proposed approach can recognise highly interacting features effectively by using only a few training samples.

Table 3 shows the average recognition time taken by four methods.[1] As stated in Section 3 and [17], both the RDetNet and SsdNet are one-stage approaches where features are extracted and recognised simultaneously. The average recognition time of the SsdNet is less than the RDetNet. The FeatureNet and MsvNet, however, are two-stage approaches, that will slow down the recognition performances [17].

4.3. Effects of proposed techniques

As noted, the RDetNet adopts a novel neural network architecture and several data augmentation (DA) strategies.[2] To fully examine the benefits of the aforementioned strategies, 16 machine learning models are trained under the following settings: (1) The RDetNet is trained under the near-optimal settings as suggested in Section 4.2. 32 models per class are randomly selected from the isolated feature dataset. DA strategies are employed to increase the diversity of the selected 3D isolated feature models. (2) To evaluate the benefits of the proposed DA strategies, this setting adopts the technique in [16] for DA. (3) This setting is identical to the 1st setting except that the DA technique is disabled. (4)–(6) As illustrated in Section 3.2.1, this paper modified the architecture of RefineDet [18] to allow for outputting 3D feature bounding boxes based on 2D viewing images. To examine the benefits of such a modification, the RDetNet that produces a set of 2D boxes rather than 3D boxes of machining features is utilised. Such a configuration is the same to the original RefineDet [18]. The depth of a 3D machining feature is estimated based on a heuristic method as done in [16]. The DA configurations in these three settings are identical to those in the first three settings. (7) The RDetNet is trained under the optimal settings, where 512 models per class are selected from the isolated feature dataset for training. The proposed DA technique is enabled for training set creation. (8) This setting is identical to the previous one except that the DA is disabled. (9)–(10) The two settings are identical to the 7th and 8th settings except that the RDetNet that produces 2D feature boxes is adopted. (11)–(12) To examine whether

[1] Please note that the authors of this paper modified the source code of the MsvNet and SsdNet to speed up the feature recognition process. Therefore, the results reported here is slight better than those reported in [17].

[2] Please note that the DA strategies mentioned here refer to the proposed strategies utilised at the training set creation stage.

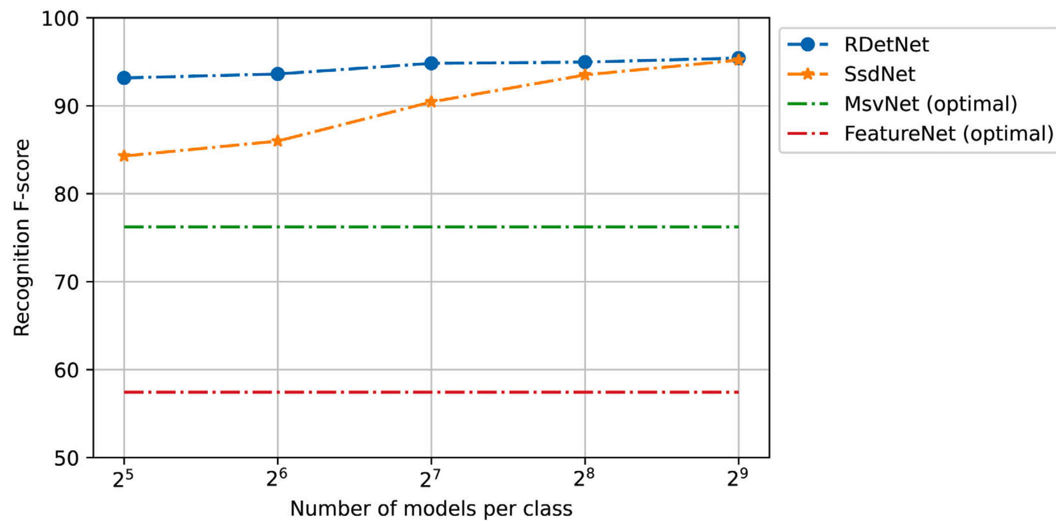


Fig. 9. Recognition F-scores (%) achieved from four approaches.

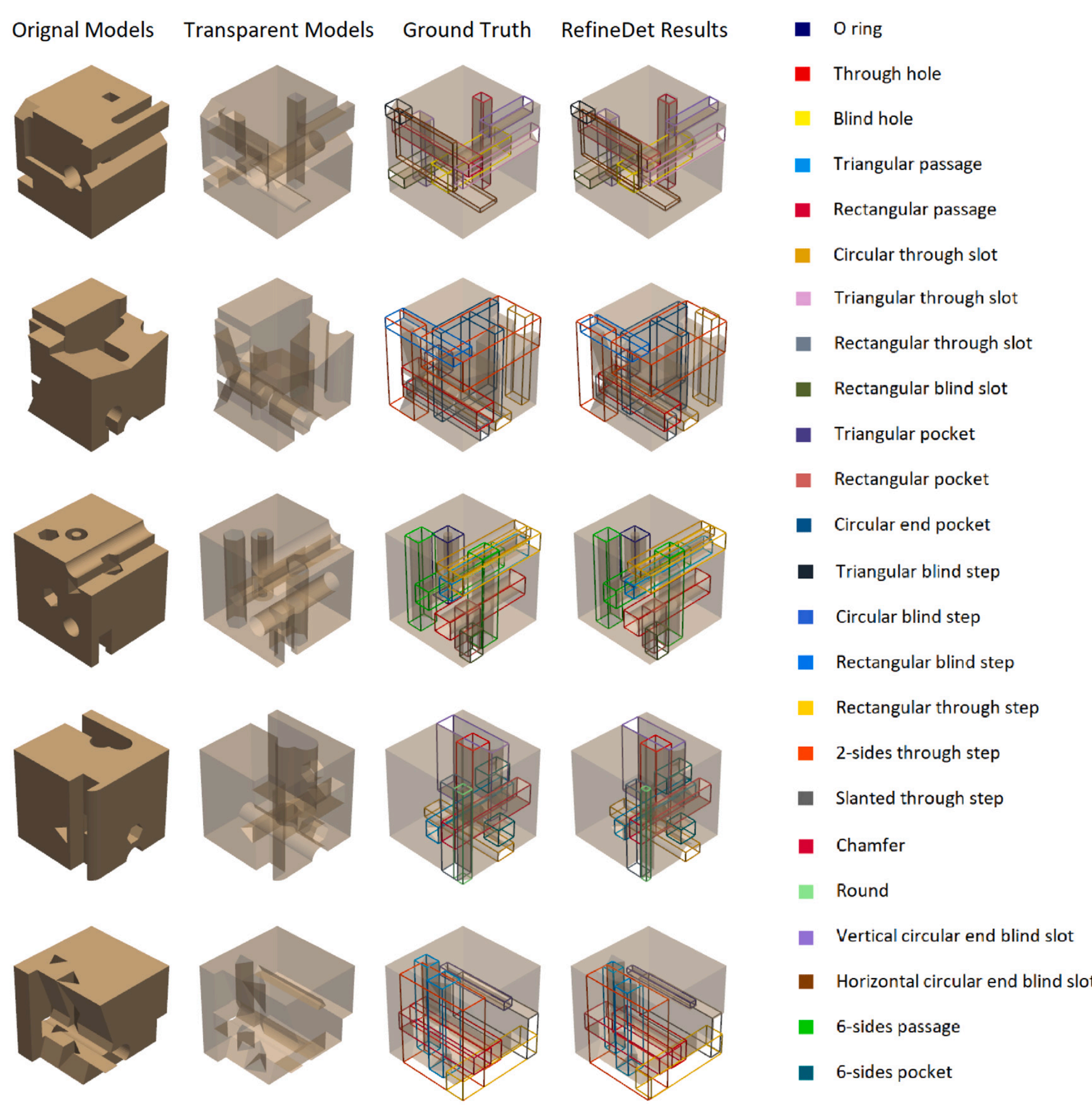


Fig. 10. Feature recognition results yield by the RDetNet trained under the sub-optimal setting.

Table 4
Experimental results (%) yielded by approaches trained under different settings.

	Method	N_{train}	DA	Output	$\mathcal{F}$
(1)	RDetNet	32	✓	3D	93.17
(2)	RDetNet	32	[16]	3D	89.53
(3)	RDetNet	32		3D	87.97
(4)	RDetNet	32	✓	2D	88.14
(5)	RDetNet	32	[16]	2D	85.84
(6)	RDetNet	32		2D	81.39
(7)	RDetNet	512	✓	3D	95.43
(8)	RDetNet	512		3D	95.38
(9)	RDetNet	512	✓	2D	90.00
(10)	RDetNet	512		2D	89.77
(11)	RDetNet	16	✓	3D	88.97
(12)	RDetNet	8	✓	3D	84.99
(13)	SsdNet	32	–	–	84.29
(14)	SsdNet	512	–	–	95.20
(15)	MsvNet	512	–	–	76.24
(16)	FeatureNet	512	–	–	57.45

the proposed method is still applicable when there are extremely small number of training samples available, the number of models per feature class utilised for training is set as 16 and 8 respectively. (13)–(16) This paper also employs the SsdNet, FeatureNet and MsvNet trained under the optimal configurations as benchmarks.

Table 4 shows the machining feature recognition results yielded by 16 machine learning models trained under the aforementioned settings. From the configuration (1) and (3), it is observed that the performances of the RDetNet have been largely enhanced by the proposed DA strategies as these strategies could increase the amount of training samples. This phenomenon, however, is less observable from the configuration (7) and (8) since there are a plenty of training samples in these configurations. Therefore, the DA strategies cannot further increase the diversity of the training set. It is notable from configuration (1) and (2) that the proposed DA technique outperforms the DA technique in [16] in terms of recognition F-score. This phenomenon is also observable from the configuration (4) and (5). It is observed from the configuration (1) and (4) that the RDetNet which produces 3D feature boxes is better than the network that produces 2D boxes. This phenomenon implies that the modification of the RefineDet is capable of enhancing the feature recognition performances. It is evident from the configuration (1) and (7) that the proposed approach could yield near-optimal results by using small training samples. From the configuration (11), (13), (15) and (16), it is notable that the RDetNet with extremely small N_{train} produces better recognition results than the FeatureNet and MsvNet trained under the optimal settings, and also outperforms the SsdNet trained using 32 STL models per class. As shown in configuration (12), the RDetNet trained using only 8 STL models per class could still produce good recognition result, which means that the proposed method is still applicable even when extremely small samples are available for training. In configuration (3) and (13), the proposed DA strategies are not adopted. It is clear that the RDetNet still produces better results than the SsdNet as the ARM adopted in the RDetNet is capable of reducing the search space and providing good initialisation. From the configuration (1) and (13), it is evident that the RDetNet yields much better result than the SsdNet by using small training samples. Based on the results achieved from the configuration (1), (15) and (16), the RDetNet trained under the near-optimal configuration outperforms MsvNet and FeatureNet trained under the optimal configurations.

5. Conclusion

This paper proposes a novel one-stage small sample learning method named RDetNet, to recognise highly interacting machining features. A number of DA strategies are utilised to increase the amount and diversity of training models. It is evident to conclude that the network architecture in the RDetNet can improve the recognition performances, as the ARM in the RDetNet is capable of reducing the search space and providing good initialisation for feature recognition. Reported results demonstrate that the proposed approach could yield near-optimal recognition results by using only a few 3D isolated feature models for training (i.e. 32 models per class), which make it easier to use in a real world scenario. The experiments in this paper are conducted based on two synthetic datasets [15,16], which are randomly constructed by scripts. In a real world scenario, the proposed network trained using real CAD models is still able to produce expected results if the real CAD parts and the STL models in this paper have similar nature (e.g. cube-like CAD models, regular rather than free-form features). In the presence of other types of features, however, some modifications to the proposed method might be required. When recognising features in an axisymmetric part, for instance, the input of neural network needs to be adjusted as the original input (i.e. viewing images) may not contain enough information about the axisymmetric features. In future work, this approach will be further extended to employ real CAD part for training, and to recognise free-form features [41] in additive manufacturing.

CRediT authorship contribution statement

Peizhi Shi: Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Software. **Qunfen Qi:** Conceptualization, Funding acquisition, Writing – review & editing. **Yuchu Qin:** Conceptualization, Writing – review & editing. **Paul J. Scott:** Funding acquisition, Writing – review & editing, Supervision. **Xiangqian Jiang:** Funding acquisition, Writing – review & editing, Supervision.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.rcim.2021.102260.

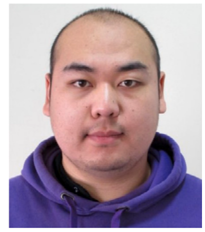
Acknowledgements

This work was supported by the EPSRC UKRI, United Kingdom Innovation Fellowship [grant number EP/S001328/1], EPSRC Future Advanced Metrology Hub [grant number EP/P006930/1] and EPSRC, United Kingdom Fellowship in Manufacturing [grant number EP/R024162/1].

References

[1] X. Xu, L. Wang, S.T. Newman, Computer-aided process planning–A critical review of recent developments and future trends, Int. J. Comput. Integr. Manuf. 24 (1) (2011) 1–31.
[2] B. Babic, N. Nesic, Z. Miljkovic, A review of automated feature recognition with rule-based pattern recognition, Comput. Ind. 59 (4) (2008) 321–337.
[3] B. Babic, N. Nesic, Z. Miljkovic, Automatic feature recognition using artificial neural networks to integrate design and manufacturing: Review of automatic feature recognition systems, (AI EDAM) Artif. Intell. Eng. Des. Anal. Manuf. 25 (3) (2011) 289.
[4] J. Han, M. Pratt, W.C. Regli, Manufacturing feature recognition from solid models: a status report, IEEE Trans. Robot. Autom. 16 (6) (2000) 782–796.
[5] X. Xu, Integrating advanced computer-aided design, manufacturing, and numerical control, Inf. Sci. Ref. (2009).
[6] Y. Shi, Y. Zhang, R. Harik, Manufacturing feature recognition with a 2D convolutional neural network, CIRP J. Manuf. Sci. Technol. 30 (2020) 36–57.
[7] Y. Zhang, X. Li, L. Gao, L. Wang, L. Wen, Imbalanced data fault diagnosis of rotating machinery using synthetic oversampling and feature learning, J. Manuf. Syst. 48 (2018) 34–50.
[8] Y. Qin, Q. Qi, P. Shi, P.J. Scott, X. Jiang, Automatic generation of alternative build orientations for laser powder bed fusion based on facet clustering, Virtual Phys. Prototyp. 15 (3) (2020) 307–324.
[9] Z. Li, Z. Zhang, J. Shi, D. Wu, Prediction of surface roughness in extrusion-based additive manufacturing with machine learning, Robot. Comput.-Integr. Manuf. 57 (2019) 488–495.

[10] H. Liang, X. Wen, Y. Liu, H. Zhang, L. Zhang, L. Wang, Logistics-involved qos-aware service composition in cloud manufacturing with deep reinforcement learning, Robot. Comput.-Integr. Manuf. 67 (2021) 101991.
[11] Z. Zhou, L. Li, A. Fürsterling, H.J. Durocher, J. Mouridsen, X. Zhang, Learning-based object detection and localization for a mobile robot manipulator in SME production, Robot. Comput.-Integr. Manuf. 73 (2022) 102229.
[12] Q. Ji, M. Chen, X.V. Wang, L. Wang, L. Feng, Optimal shape morphing control of 4D printed shape memory polymer based on reinforcement learning, Robot. Comput.-Integr. Manuf. 73 (2022) 102209.
[13] A. Al-Yacoub, Y. Zhao, W. Eaton, Y. Goh, N. Lohse, Improving human robot collaboration through Force/Torque based learning for object manipulation, Robot. Comput.-Integr. Manuf. 69 (2021) 102111.
[14] R. Zhang, Q. Lv, J. Li, J. Bao, T. Liu, S. Liu, A reinforcement learning method for human-robot collaboration in assembly tasks, Robot. Comput.-Integr. Manuf. 73 (2022) 102227.
[15] Z. Zhang, P. Jaiswal, R. Rai, Featurenet: Machining feature recognition based on 3D convolution neural network, Comput. Aided Des. 101 (2018) 12–22.
[16] P. Shi, Q. Qi, Y. Qin, P.J. Scott, X. Jiang, A novel learning-based feature recognition method using multiple sectional view representation, J. Intell. Manuf. 31 (5) (2020) 1291–1309.
[17] P. Shi, Q. Qi, Y. Qin, P. Scott, X. Jiang, Intersecting machining feature localization and recognition via single shot multibox detector, IEEE Trans. Ind. Inf. 17 (5) (2021) 3292–3302.
[18] S. Zhang, L. Wen, X. Bian, Z. Lei, S.Z. Li, Single-shot refinement neural network for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4203–4212.
[19] Y. Shi, Y. Zhang, K. Xia, R. Harik, A critical review of feature recognition techniques, Comput.-Aided Des. Appl. 17 (5) (2020) 861–899.
[20] J.H. Vandenbrande, A.A. Requicha, Spatial reasoning for the automatic recognition of machinable features in solid models, IEEE Trans. Pattern Anal. Mach. Intell. 15 (12) (1993) 1269–1285.
[21] S. Joshi, T.-C. Chang, Graph-based heuristics for recognition of machined features from a 3D solid model, Comput. Aided Des. 20 (2) (1988) 58–66.
[22] P.K. Venuvinod, S. Wong, A graph-based expert system approach to geometric feature recognition, J. Intell. Manuf. 6 (3) (1995) 155–162.
[23] S. Gao, J.J. Shah, Automatic recognition of interacting machining features based on minimal condition subgraph, Comput. Aided Des. 30 (9) (1998) 727–739.
[24] T.C. Woo, Feature extraction by volume decomposition, in: Proceedings of Conference on CAD/CAM Technology in Mechanical Engineering, Vol. 76, 1982.
[25] Y. Woo, Fast cell-based decomposition and applications to solid modeling, Comput. Aided Des. 35 (11) (2003) 969–977.
[26] M.G. Marchetta, R.Q. Forradellas, An artificial intelligence planning approach to manufacturing feature recognition, Comput. Aided Des. 42 (3) (2010) 248–256.
[27] V. Rameshbabu, M. Shunmugam, Hybrid feature recognition method for setup planning from STEP AP-203, Robot. Comput.-Integr. Manuf. 25 (2) (2009) 393–408.
[28] T. Dipper, X. Xu, P. Klemm, Defining, recognizing and representing feature interactions in a feature-based data model, Robot. Comput.-Integr. Manuf. 27 (1) (2011) 101–114.
[29] M. Al-wswasi, A. Ivanov, A novel and smart interactive feature recognition system for rotational parts using a STEP file, Int. J. Adv. Manuf. Technol. (2019) 1–24.
[30] N. Ismail, N.A. Bakar, A. Juri, Feature recognition patterns for form features using boundary representation models, Int. J. Adv. Manuf. Technol. 20 (8) (2002) 553–556.
[31] Q. Wang, X. Yu, Ontology based automatic feature recognition framework, Comput. Ind. 65 (7) (2014) 1041–1052.
[32] Y. Zhang, X. Luo, B. Zhang, S. Zhang, Semantic approach to the automatic recognition of machining features, Int. J. Adv. Manuf. Technol. 89 (1–4) (2017) 417–437.
[33] S. Meeran, A. Zulkifli, Recognition of simple and complex interacting non-orthogonal features, Pattern Recognit. 35 (11) (2002) 2341–2353.
[34] K. Nezis, G. Vosniakos, Recognizing 212D shape features using a neural network and heuristics, Comput. Aided Des. 29 (7) (1997) 523–539.
[35] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, Ssd: Single shot multibox detector, in: European Conference on Computer Vision, Springer, 2016, pp. 21–37.
[36] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[37] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[38] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[39] N. Bodla, B. Singh, R. Chellappa, L.S. Davis, Soft-NMS–Improving object detection with one line of code, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 5561–5569.
[40] A.C. Wilson, R. Roelofs, M. Stern, N. Srebro, B. Recht, The marginal value of adaptive gradient methods in machine learning, in: Advances in Neural Information Processing Systems, 2017, pp. 4148–4158.
[41] S. Bendjebla, N. Cai, N. Anwer, S. Lavernhe, C. Mehdi-Souzani, Freeform machining features: new concepts and classification, Procedia CIRP 67 (2018) 482–487.



Peizhi Shi is currently a research fellow at the EPSRC Future Advanced Metrology Hub, University of Huddersfield, UK. He received his Ph.D. in computer science from the University of Manchester in 2019, master's degree in software engineering from the University of Science and Technology of China in 2013, and bachelor's degree in computer science from the Guilin University of Electronic Technology in 2010. His current research interests include machine learning, 3D object detection, pattern recognition, machine perception and their applications in intelligent manufacturing.

Qunfen Qi is currently a senior research fellow at the EPSRC Future Advanced Metrology Hub, University of Huddersfield, UK. She received a Ph.D. degree in Precision Engineering from University of Huddersfield in 2013. She is an EPSRC UKRI Innovation Fellow, an EPSRC Peer Review Full College Member, an EPSRC Women in Engineering Society (WES) member, and a fellow of the Higher Education Academy (FHEA). Her research lies in knowledge modelling for manufacturing covering smart information systems, abstract mathematical theory (category theory), geometrical product specifications (GPS), additive manufacturing (AM), and surface metrology. She has worked for fifteen years in developing decision-making tools for smart product design and inspection, using category theory as its foundation.

Yuchu Qin is a research fellow with the EPSRC Future Advanced Metrology Hub, University of Huddersfield, Huddersfield, UK. He received the second Ph.D. in advanced manufacturing and precision engineering with the EPSRC Future Advanced Metrology Hub, University of Huddersfield, Huddersfield, UK in 2021, the first Ph.D. in measurement technology and instrument from the School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, China in 2017, and the M.Eng. in computer application technology and the B.Eng. in computer science and technology from the School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin, China in 2013 and 2010, respectively. His research interests include intelligent manufacturing, computational intelligence, and knowledge engineering.

Paul J. Scott is currently a professor at the EPSRC Future Advanced Metrology Hub of the University of Huddersfield. He received a Ph.D. degree in Statistics from Imperial College London in 1983. He has an honours degree in Mathematics and an M.Sc. degree in Statistics. His research interests are in manufacturing informatics, geometrical product specifications and verification, philosophy of the measurement of product geometry, and foundations of specifying and characterising solutions for real world industrial problems. He was the project leader for twenty published ISO standards and is currently working on four new ISO documents. He is a fellow of Royal Statistical Society (FRSS), an EPSRC Fellow of Manufacturing, a leading member of ISO TC 213, a founder member of the strategic group AG1 and the technical review group AG2 of ISO TC 213, a convenor of the working group WG15 (Filtration and Extraction) and the advisory group AG12 (Mathematics for Geometrical Product Specifications) of ISO TC 213, a core member of BSI TDW4, a convenor of BSI TDW4/-/9, a visiting industrial professor of Taylor Hobson Ltd, and the Taylor Hobson Chair for Computational Geometry.

Xiangqian Jiang is currently the chair professor and the director of the EPSRC Future Advanced Metrology Hub, University of Huddersfield and the Royal Academy of Engineering and Renishaw Chair in Precision Metrology. She has a D.Sc. degree in Precision Engineering and a Ph.D. degree in Surface Metrology. Her research interests mainly lie in Surface Measurement, Precision Engineering, and Advanced Manufacturing Technologies. She was made a Dame Commander (DBE) of the Order of the British Empire for services to Engineering and Manufacturing in 2017. She is a fellow of the Royal Academy of Engineering (FREng), a fellow of the Royal Society of Arts (FRSA), a fellow of the Institute of Engineering Technologies (FIET), a fellow of the International Academy of Production Research (FCIRP), a fellow of the International Society for Nanomanufacturing (FISNM), a principle member of ISO TC 213 and BSI TW/4, an advisory member for UK national measurement system, and the UK Chairman of the International Academy of Production Research.