

Automated manufacturability analysis and machining process selection using deep generative model and Siamese neural networks

Xiaoliang Yan, Shreyes Melkote*

George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332, United States

ARTICLE INFO

Keywords:

Generative machine learning
Deep metric learning
Computer aided process planning
Machining feature recognition

ABSTRACT

Industry 4.0 calls for highly autonomous manufacturing process planning. Significant effort has been devoted over the years to generative Computer Aided Process Planning (CAPP), which aims to generate process plans for new designs without human intervention. This goal has not been realized to date due to several reasons, such as poor scalability and the difficulty in modeling manufacturing process capability, which encapsulates the part shape, quality, and material property transformation capabilities of the process. In our prior work, the shape transformation capabilities of lathe-based machining operations were modeled as latent probability distributions using a data-driven deep learning-based generative machine learning approach, from which visualizations of realistic machinable features could be sampled to assist *manual* process selection. In this paper, a Siamese Neural Network (SNN) is integrated with Autoencoder-based deep generative models of machining operations to enable automated comparison of the query part shapes with sampled outputs. This enables automated manufacturability analysis and machining process selection necessary for generative CAPP. The paper also demonstrates that the proposed Autoencoder and Siamese Neural Network (AE-SNN) achieves a class-average process selection accuracy of 89 %, and a manufacturability analysis accuracy of 100 %, which outperforms a discriminative model trained on the same dataset.

1. Introduction

Process planning is a critical decision-making step in discrete part manufacturing that results in a detailed plan necessary for producing a part from a specified design [1]. Industry 4.0 calls for highly automated process planning to bridge the long-standing gap between Computer-Aided Design (CAD) and manufacturing [2]. Significant efforts have been devoted to Computer Aided Process Planning (CAPP) since its inception in the 1960s, but few computerized systems have reached the level of automation envisioned by researchers [3].

Two predominant approaches to CAPP, variant and generative, were proposed in the 1980s [4]. Given a new part design, variant CAPP utilizes Group Technology methods [5] to search for a similar part shape in its database of previously manufactured parts. The generic process plan associated with the similar part is then manually edited to meet the new part design requirements. In contrast, generative CAPP aims to automatically create process plans for new designs without the need for human intervention. However, this approach has been hindered by limitations of rule-based expert systems commonly employed in prior work [6]. Although variant CAPP is still prevalent in practice,

according to Xu et al. [7], recent advances in modern machine learning have renewed the interest in solving the challenges of generative CAPP. Prior works have developed generative CAPP systems for process selection [8,9], operations sequencing [10,11], and machine setup planning [12,13]. However, surveys show that most works on CAPP are not easily scalable to other processes, and no truly generative CAPP system has been realized to date [3].

A major research area in generative CAPP is focused on feature recognition (FR) from CAD models. The initial motivation for FR was to integrate CAPP with CAD for machining [3]. FR techniques such as syntactic pattern recognition [14], graph theory [15], hint-based recognition [16], expert systems [17], and neural networks [18] have been proposed and have achieved varying levels of success. However, these methods have limitations such as failure to recognize interacting features, or missing design details. In addition, traditional FR systems, e.g., graph-based systems, assume that a recognized feature can always be manufactured, without explicitly evaluating whether there exists a manufacturing process capable of producing the query features [19]. While manufacturability of a discrete part depends on many factors such as shape, workpiece material, geometric dimensioning and toler-

* Corresponding author.

E-mail address: shreyes.melkote@me.gatech.edu (S. Melkote).

<https://doi.org/10.1016/j.jmsy.2023.01.006>

Received 26 September 2022; Received in revised form 15 December 2022; Accepted 9 January 2023
0278-6125/© 20XX

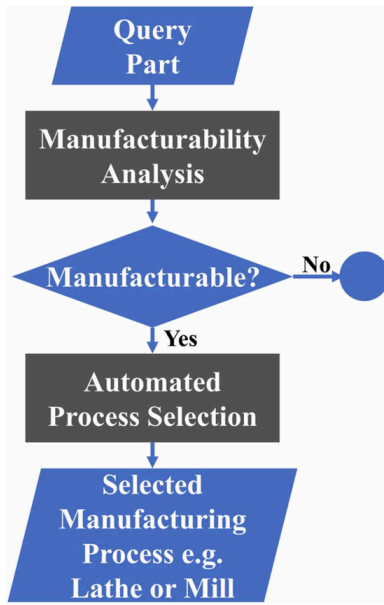


Fig. 1. Envisioned workflow for automated manufacturability analysis and process selection.

ancing (GD&T), and availability of required manufacturing resources, the primary attention, especially in the context of FR, has been focused on shape and geometric attributes of features in determining manufacturability of a query part.

Recent developments in 3-dimensional (3D) machine learning have renewed interest in FR using data-driven methods. Zhang et al. [20] proposed FeatureNet that utilized a 3D convolutional neural network (3D-CNN) to automatically identify features present in a voxelized query part. Ghadai et al. [21] presented an approach to localize and identify drilled holes and analyze their manufacturability based on known manufacturability rules. Peddireddy et al. [22] developed a machining process identification system leveraging transfer learning from a pre-trained basic feature recognition model. Fu et al. [23] proposed Improved Dixel Representation (IDR) for CAD models that improved the classification accuracy of manufacturing processes using CNN.

Wang and Rosen [24] developed a manufacturing process classification method using Heat Kernel Signature and CNN. Ning et al. [25] proposed an improved machining feature recognition approach that combines a graph-based approach and 3D-CNN. It is evident that increasing emphasis has been placed on bridging the gap between FR and manufacturability analysis and process selection, but current state-of-the-art classification-based methods tend to suffer from the intrinsic limitations of discriminative machine learning techniques, which only perform classification in an implicit manner as a “black-box” [3]. Since a part can often be produced by more than one machining process, the multi-label classification task requires large numbers of training classes consisting of both machinable and non-machinable features. Although efforts to classify the manufacturability of a query part have been reported [22], the need for non-machinable feature data renders this approach difficult to implement in practice.

An alternative modeling approach is deep generative modeling, which has been proposed to solve the lack of training data in other disciplines such as industrial internet of things (IIoT.) Specifically, deep generative models have demonstrated success in anomaly detection and trust-boundary protection without a large anomaly dataset [26], which is analogous to manufacturability analysis in process planning. For instance, Jiang et al. [27] proposed a Generative Adversarial Network (GAN)-based anomaly detection approach for industrial time-series data, which aims to detect anomalies without prior knowledge of abnormal samples. Belenko et al. [28] presented an anomaly detection approach that uses GAN for large-scale wireless communication of cyber physical systems. Hassan et al. [29] proposed an adaptive trust boundary protection layer that utilizes a deep-learning feature-extraction-based neural network that is resilient to unlabeled modes of cyberattack. Wang et al. [30] developed a Recurrent Neural Network-based Encoder-Decoder-Attention network for anomaly detection in manufacturing time-series data. Yan and Yan [31] presented an Adversarial Autoencoder coupled with a Fully Connected Neural Network for manufacturing quality fault detection. Benefits of utilizing generative models have also been demonstrated in design topology optimization, where the high computational cost of Finite Element Analysis has led to the use of deep generative models such as Variational Autoencoder (VAE) and GAN to generate near-optimal topological designs [32]. Banga et al. [33] employed a 3D Autoencoder (AE) to optimize a structural design. Greninger [34] presented a 3D-GAN that enforces manufacturing

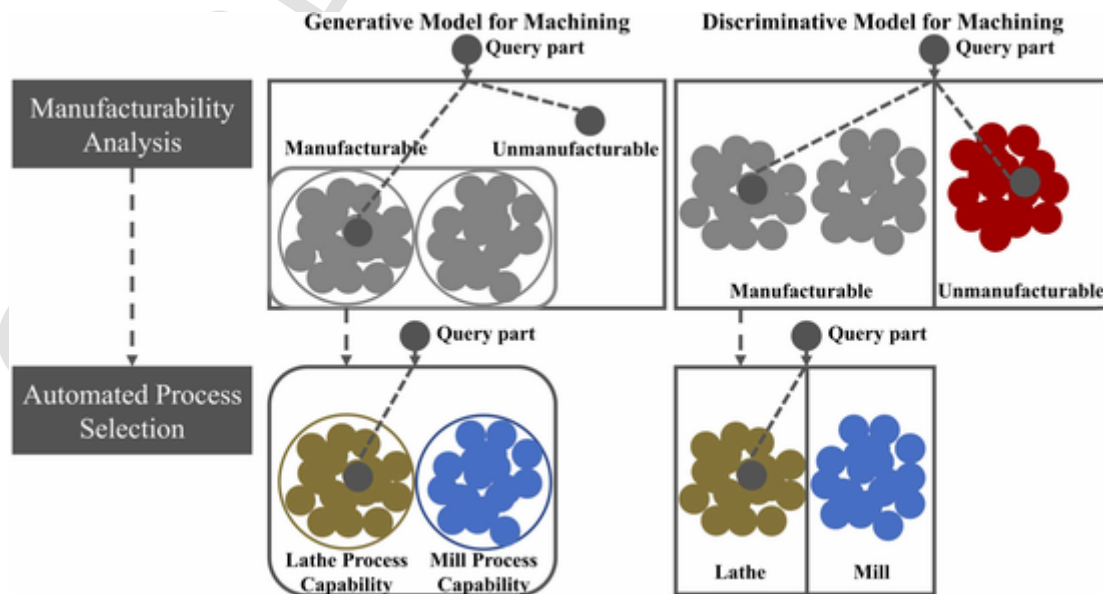


Fig. 2. Differences between generative and discriminative modeling approaches for manufacturability analysis and process selection (right). Machining is used as an example.

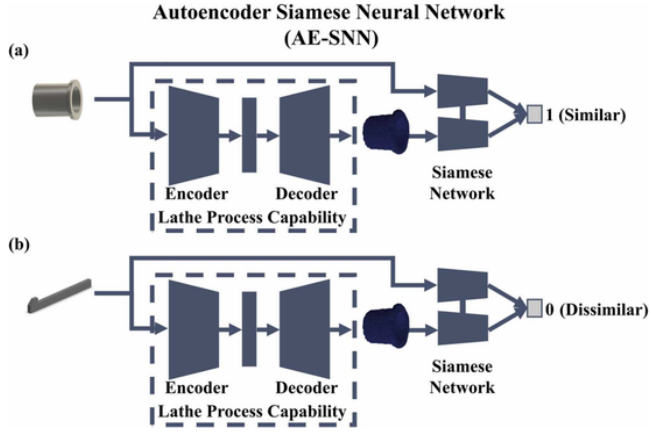


Fig. 3. Workflow of AE-SNN with use case examples: (a) the query part shape and the part shape output by the lathe-based process capability model are similar (denoted by 1), which implies that the query part is manufacturable by lathe-based machining operations, and (b) the query part is not similar (denoted by 0) to the output of the lathe-based process capability, implying that the query part cannot be manufactured by lathe-based operations.

constraints on topology optimization by training with known shapes manufactured by a 3-axis milling machine. Hertlein et al. [35] proposed a conditional GAN in early stage topology optimization for additive manufacturing.

In light of the above, in our prior work we employed a generative model to learn the shape transformation capability of simple lathe-based machining operations [36]. Deep generative models of machining operations were used to learn their shape transformation capability to enable qualitative visualizations that can assist designers in manufacturability assessment as well as process planners in process selection. In this work, we extend our previous generative modeling work beyond qualitative assessment by proposing a deep metric learning-based quantitative method that concurrently enables automated manufacturability assessment and automated process selection. Here, we limit the scope of the manufacturability analysis and process selection to only the shape and geometric attributes of a query part, but the methodology pre-

sented can be extended to include other factors such as material and quality attributes. Specifically, we build on our prior work in generative modeling of the shape transformation capability of machining processes to answer the following questions: (1) Can the proposed deep metric learning approach enable automated manufacturability analysis and process selection? and (2) How well does the quantitative metric perform in manufacturability assessment and process selection compared to a deep discriminative model? Fig. 1 envisions the workflow of automated manufacturability analysis and process selection, where the manufacturability of a query part is first determined before a manufacturing process is selected. These questions are answered by developing and evaluating a combined Autoencoder and Siamese Neural Network (AE-SNN). The performance of the AE-SNN in manufacturability assessment and process selection is evaluated through a direct comparison with the performance of a baseline 3D-CNN classification model.

2. Modeling approach

2.1. Generative modeling of the shape transformation capability of machining processes

In our prior work [36], we trained three 3D-VAE-GAN based generative models to learn the shape transformation capabilities of representative lathe-based machining operations, namely turning, grooving, and chamfering. Using this modeling approach, we demonstrated that the shape transformation capability of a machining process can be learned as a latent probability distribution, from which machinable features can be visualized through sampling. Specifically, while a discriminative model learns the conditional probability $P(y|X)$ from a training dataset, where, in the context of machining, X are the data points, such as part designs, and y are the corresponding labels, such as machining process/operation labels, a generative model of the shape transformation capability of the machining process learns the underlying joint probability $P(X, y)$ from the dataset, or $P(X)$ if labels are not available. As notionally illustrated in Fig. 2, generative modeling learns the underlying manufacturing process capability, which can be first utilized to analyze manufacturability of a part by determining if the part requirements (e.g., shape) can be met by existing process capabilities. Subsequently, the generative model is used to select a machining process that can pro-

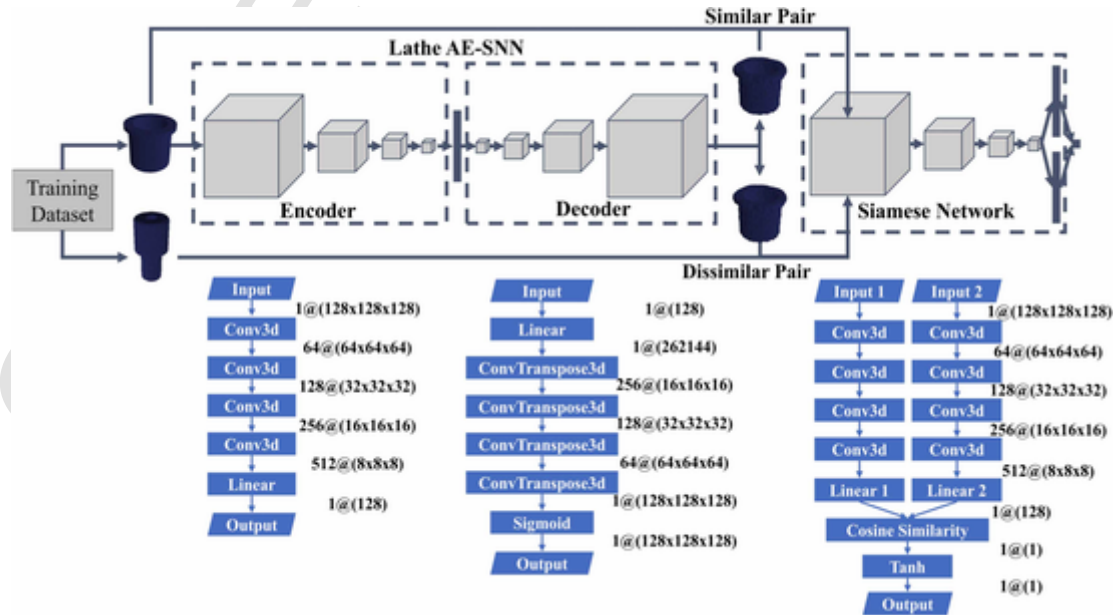


Fig. 4. AE-SNN architecture and training process. During training, the output of the AE is used in conjunction with the input to the AE as positive pair inputs to the Siamese Network. The same output of the AE is used in conjunction with another randomly drawn sample from the training dataset, which is not used in the AE, as negative pair inputs to the Siamese Network.

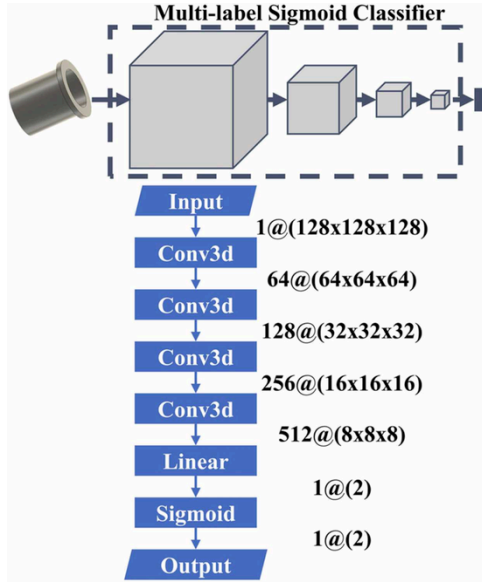


Fig. 5. 3D-CNN discriminative model with Sigmoid activation function.

duce the query part. In contrast, discriminative modeling learns the class boundaries from a training dataset and outputs the class that the query part belongs to; however, manufacturability analysis using discriminative methods require additional training data representing unmanufacturable parts (depicted in red in Fig. 2), which are generally not available in practice.

However, a limitation of our prior 3D-VAE-GAN based generative approach is that a human must be incorporated into the computational loop to *manually* assess manufacturability and perform process selection through visualization. In contrast, to enable automated manufacturability assessment and process selection, a quantitative metric is necessary. A deep generative metric learning approach is presented in this paper to automatically evaluate the similarity of the part shapes output by the generative models of machining processes to the query part shapes.

2.2. Deep generative metric learning approach

Recall that the part shapes output by a trained generative shape transformation capability model are sampled from a learned latent probability embedding of machinable parts. By evaluating the similarity between the input and output shapes, manufacturability of a query part can be established. Automatic evaluation of 3D shapes generated by an Autoencoder (AE) in response to an input shape requires the ability to learn a similarity metric that can distinguish between the output and input shapes. While numerous voxel-based similarity search methods using volume, solid angle, eigenvalues, and nearest neighbor clustering have been proposed [37], the desired similarity threshold must be hard-coded to satisfy a specific application. Alternatively, a data-driven similarity metric [38] can be trained to judge the similarity between the input and output of a generative model. In this paper, a Siamese neural network (SNN) [39] is trained in conjunction with an AE model to determine the similarity between the input and output of the AE.

SNNs have been utilized in applications such as one-shot image recognition, facial recognition, additive manufacturing [40], and learning from demonstration in robotics [41]. Their twin-network architecture encodes two objects simultaneously and computes their difference to evaluate similarity. In this work, we employ an AE-SNN framework to learn the shape transformation capability of typical machining operations. Specifically, for the purposes of this paper, we consider the problem of learning the shape transformation capability of lathe-based machining operations such as turning, grooving, and chamfering and machining operations carried out on a milling machine (or mill). The AE-SNN was trained on a dataset consisting of manufacturable parts only. As shown in Fig. 3, once the AE-SNN model is trained, whether a query part falls into the general range of process capabilities of the lathe or the mill is determined by the similarity score between the input query part and the outputs of the respective process capability models.

2.3. Model architecture

Fig. 4 depicts the model architecture of the AE-SNN and the training process. Specifically, convolutional neural networks with roles of “encoder,” “decoder,” and “Siamese network” were trained. The encoder takes the voxelized ground truth shapes from the training dataset

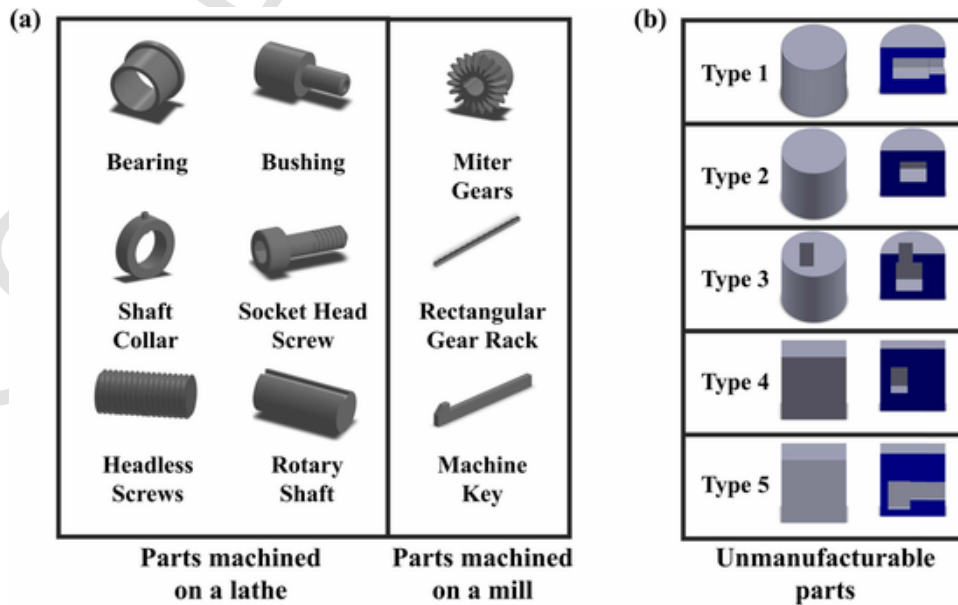


Fig. 6. Datasets for (a) manufacturable parts categorized by the machining process/operation used to produce them, and (b) unmanufacturable parts that cannot be produced on a lathe or on a mill.

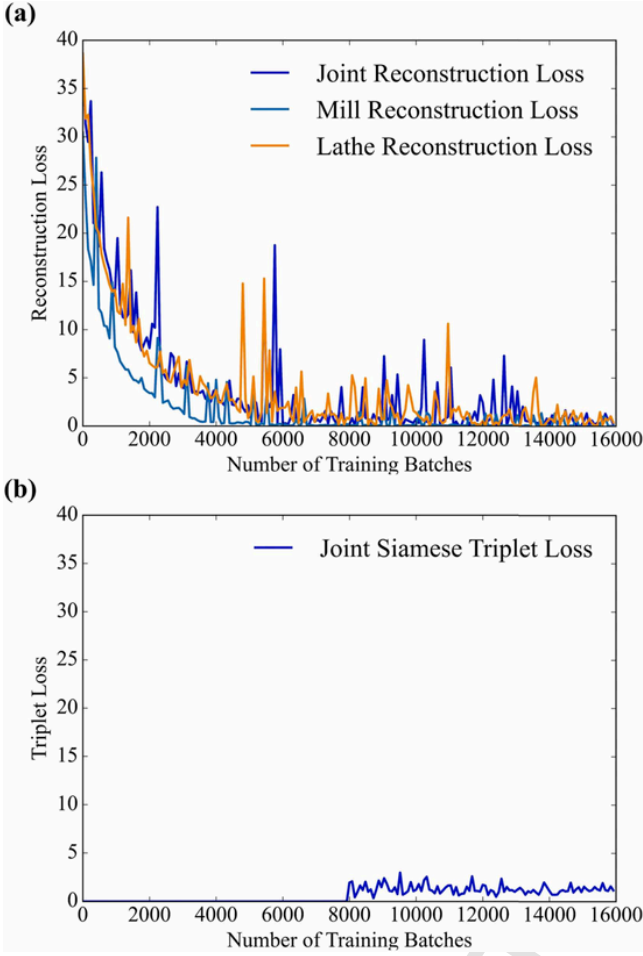


Fig. 7. Training losses (a) AE reconstruction loss, and (b) SNN triplet loss.

consisting of 3D CAD models of parts manufacturable on a lathe (or mill) as input to learn the corresponding latent probability distribution of the shape transformation capability of the process/operation, while the decoder generates realistic features that can be produced on a lathe (or a mill), and the SNN judges the “similarity” between the input and output. Each voxelized input object drawn from the training dataset has dimensions of $128 \times 128 \times 128$. The encoder has four 3D-convolutional (Conv3d) layers and one linear layer. Batch normalization [42] and the leaky rectified linear unit (LReLU) activation function [43] are applied after each Conv3d layer. LReLU is a non-linear activation function given by $LReLU(x) = \max(x, \gamma x)$, where γ in this work is set to 0.2. The fully connected linear layer following the four Conv3d layers has dimensions of 128×1 . In the decoder, a fully connected layer expands the dimensions to $262,144 \times 1$ and is followed by four 3D-transpose-convolutional (ConvTranspose3d) layers. A Sigmoid layer, which has the form $\sigma(x) = (1 + e^{-x})^{-1}$, is applied at the end to enforce a value between 0 and 1 for each voxel in the $128 \times 128 \times 128$ output. The SNN has a network structure like the AE, followed by a cosine similarity layer and the Tanh layer, which has the form of $T(x) = (e^x - e^{-x}) / (e^x + e^{-x})$. The output is then linearly transformed by $S = (T(x) + 1) / 2$ to ensure that the output falls between 0 and 1. Both similar and dissimilar pairs of samples are used to train the SNN. In the training dataset, a similar pair is composed of the voxelized part shape input to the AE and the corresponding output generated by the decoder. A dissimilar pair is composed of the same decoder output as in the similar pair and a randomly drawn sample from the training dataset, which must be different from the encoder training input.

2.4. Objective functions

In a typical encoder and decoder (generator) pair, the loss function for both networks is defined as follows:

$$\alpha_1 \|\hat{x} - x\|_2 \quad (1)$$

where x is the encoder input, $\hat{x}$ is the generator output, and α_1 is a tuning parameter. This loss function returns the reconstruction loss between the input and output. In this work, α_1 is set to 150.

The SNN employs a triplet loss function. A training triplet comprises an anchor, a positive that is considered “similar” to the anchor (similar pair), and a negative that is dissimilar to the anchor (dissimilar pair). The objective function is given by:

$$\alpha_2 \log(1 - S(x_a, x_n)) + \log S(x_a, x_p) \quad (2)$$

where x_a , x_p , and x_n are the anchor, positive, and negative in a triplet, respectively. In this work, x_a are the generated decoder outputs, x_p are the training inputs to the AE, and x_n are the randomly drawn samples from the training dataset that are not used as inputs in the training phase. The value of α_2 in this work is set to 2.

2.5. Baseline model for comparative study

A discriminative model was constructed to serve as baseline for manufacturability analysis and process selection. The model architecture generally follows that described by Peddireddy et al. [22] for machining process and manufacturability classification, and is shown in Fig. 5. The Sigmoid activation function was implemented for multi-label classification. Since we limit the number of candidate machining processes to two (lathe- and mill-based), the output is then a vector of size 2×1 with each element of the output vector representing either a lathe-based or a mill-based manufacturing process/operation. The output vector indicates whether the query part can be manufactured by one process ((1, 0) or (0, 1)), two processes (1, 1), or if the query part is unmanufacturable (0, 0).

3. Model training and evaluation

3.1. Data curation and processing

Despite recent advances in 3D machine learning, 3D datasets for manufacturing are scarce. As mentioned earlier, in this work we limit our focus to machining processes/operations that can be performed on a lathe or a mill.

To evaluate manufacturability analysis and to subsequently enable automated process selection, a training dataset consisting of manufacturable parts with manufacturing process labels must be created. To this end, we curated and processed publicly available 3D data from the FabWave CAD repository [44], which was categorized by researchers at North Carolina State University based on functional classes (e.g. bearing, bushing, gears, etc.). As shown in Fig. 6(a), six functional classes are selected as parts that can be produced on a lathe, and three functional classes are selected as parts that can be produced on a milling machine. In all, 1235 parts machined on a lathe and 112 parts machined on a mill are collected in two corresponding datasets. While these parts can potentially be made using other manufacturing processes, we limit the scope of process selection in this work to operations that can be performed either on a lathe or a mill. The methodology, however, can be generalized to other discrete manufacturing processes.

Fig. 6(b) shows five types of unmanufacturable parts consisting of prismatic and axis-symmetric features, which were generated using the non-machinable feature generation approach described by Peddireddy et al. [22]. Specifically, 200 part models were generated

Class		Voxelized Input	Voxelized Output	Similarity
Lathe Bearing				0.9947
Lathe Bushing				0.9956
Lathe Headless Screw				0.9944
Lathe Rotary Shaft				0.9837
Lathe Shaft Collar				0.9935
Lathe Socket Head Screw				0.9954
Mill Machine Key				0.9996
Mill Miter Gear				0.9803
Mill Rectangular Gear Rack				0.9961

Fig. 8. Examples of input and output pairs from process-specific AE-SNNs.

for each type of non-machinable parts. Data generation was automated using a Solidworks macro. The CAD models obtained were subsequently voxelized with a resolution of $128 \times 128 \times 128$ using binvox [45], an open-source voxelizer library.

3.2. Model training approach

In this work, 800 lathe-based parts and 80 milled parts were used for training, and 435 lathe-based parts and 32 milled parts were used for testing. For manufacturability analysis, we trained a joint AE-SNN model on the manufacturable parts CAD dataset consisting of lathe and milled parts. For process selection, two additional AE-only models were

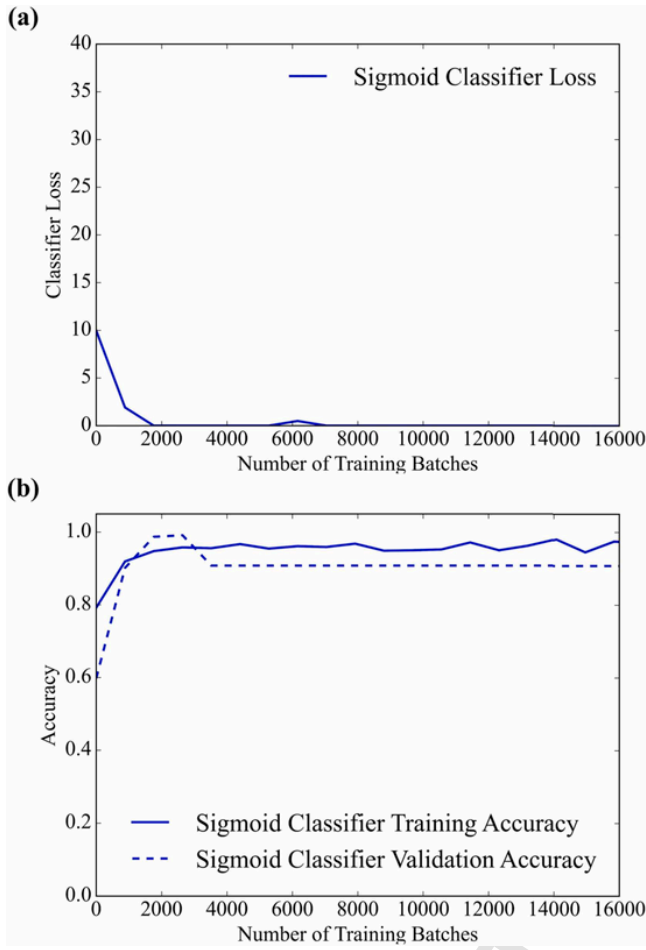


Fig. 9. Baseline Sigmoid classifier model (a) training loss, and (b) training and validation accuracies.

trained, one for each process class, and the SNN from the joint AE-SNN model was reused to provide a similarity score for the input-output pairs for the two AE-only models. The models were constructed and trained using PyTorch [46], which is a deep learning library for Python. Model training was performed on a high-performance computing node at the Georgia Institute of Technology (PACE Phoenix Cluster with 2 parallel NVIDIA Tesla V100 16 GB GPUs). Due to data imbalance between the two process classes, the models were trained for a fixed number of batches, instead of epochs. The batch size in this work is one, and each model was trained for 16,000 batches. For AE-SNN, the encoder and decoder were trained simultaneously during the first 8000 training batches while keeping the Siamese network constant. All three constituents of the AE-SNN were trained simultaneously for the remaining 8000 batches. For AE-only models, the encoder and decoder were trained simultaneously for 16,000 batches. Adaptive moment estimation (Adam) [47] was used as the optimizer with initial exponential decay rates of the first and second moments of gradient, β_1 and β_2 , set to 0.8 and 0.99, respectively. The encoder and decoder learning rates were set to 4×10^{-6} , and the Siamese network learning rate was set to 2×10^{-6} .

For comparison with a discriminative model, the Sigmoid classifier was trained on a combined lathe and milled parts dataset consisting of 880 parts. The learning rate of the discriminative model was set to 4×10^{-6} , while the other hyperparameters were kept the same as in the AE-SNN case.

3.3. Model training evaluation

The trained AE-SNN and AE-only models were evaluated by observing the training losses and the shape transformation (i.e., object generation) results. Fig. 7 shows the AE reconstruction loss given by Eq. (1), and the SNN triplet loss given by Eq. (2), of the joint AE-SNN model that was trained on both lathe and milled parts. In addition, the reconstruction losses of the mill and lathe AE-only models are shown in Fig. 7 (a). As training progresses, the reconstruction losses of all three models decrease and converge to a level close to zero after 6000 training batches. However, high volatility in the training losses can be observed, with higher volatility observed in the lathe AE-only and the joint AE-SNN models. This volatility is attributed to the small batch size, which in the present case is one. A higher batch size will lead to lower volatility in training losses but will also require a longer training time to converge to the desired loss level. Nevertheless, all three models reached low loss levels after 16,000 training batches, which suggests that the batch size of one is sufficient for training the AE-SNN and AE-only models over the given dataset. As discussed in the model training approach section, training of the SNN started after 8000 training batches after which the triplet loss quickly converged to a low level for the remaining training batches, indicating that the SNN model was able to distinguish between positive and negative pairs of training samples. In addition to the training loss, training data reconstructions were visualized, as shown in Fig. 8, to evaluate the learned shape transformation capability. The outputs of the SNN are shown as similarity scores next to the corresponding input-output pairs. A similarity score ranges between 0 and 1, with 0 indicating a dissimilar input-output pair, and 1 indicating a similar input-output pair. The high similarity scores further demonstrate that the Autoencoders have learned the shape transformation capability of the corresponding machining process.

Similarly, the baseline discriminative Sigmoid classifier model was trained and the corresponding training loss and validation accuracy evaluated as shown in Fig. 9(a) and in Fig. 9(b). It is evident that the differences in the training and validation accuracies are small, which indicates that overfitting is insignificant. Note that the accuracy shown is calculated with a class probability threshold of 0.5 to illustrate the training and validation processes. A more detailed receiver operating characteristic plot (ROC) and area under the curve (AUC) are shown in the manufacturability analysis section.

4. Comparative analysis

Comparative analysis comprises manufacturability analysis and automated process selection. An AE-SNN and a baseline sigmoid classifier are trained as described in Section 3.3 and compared. For manufacturability analysis validation, both machinable testing dataset and non-machinable dataset were used. For process selection, only the machinable testing dataset was divided into process-specific datasets and used for validation.

4.1. Manufacturability analysis

Manufacturability analysis is the first task that is possible with the AE-SNN model. Prior literature has demonstrated high accuracy of the baseline Sigmoid Classifier to determine manufacturability of a query part when an extensive unmanufacturable parts dataset is used in training [22]. In an example case where the part cannot be made by either a mill or a lathe, the model returned a tuple of (0, 0). The goal of manufacturability analysis presented in this work, however, requires deep learning models to determine the manufacturability of query parts without training on unmanufacturable parts data. This task is relevant in an industry setting where unmanufacturable parts data are not readily available. Therefore, in this work, the AE-SNN and the baseline Sigmoid Classifier were not trained on unmanufacturable parts dataset. Us-

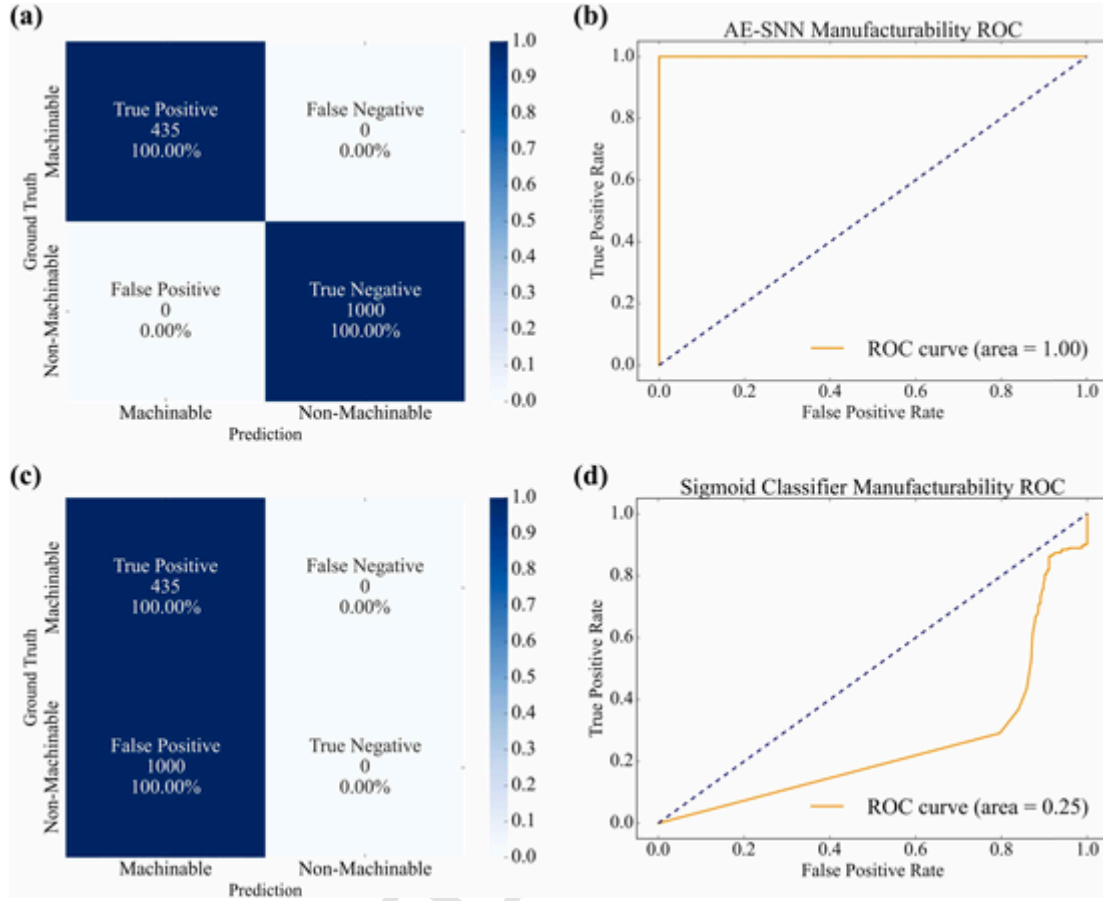


Fig. 10. Manufacturability accuracy comparison of the AE-SNN and Sigmoid Classifier models, (a) confusion matrix for the AE-SNN model, (b) AE-SNN ROC plot, (c) confusion matrix for the Sigmoid Classifier, and (d) Sigmoid Classifier ROC plot.

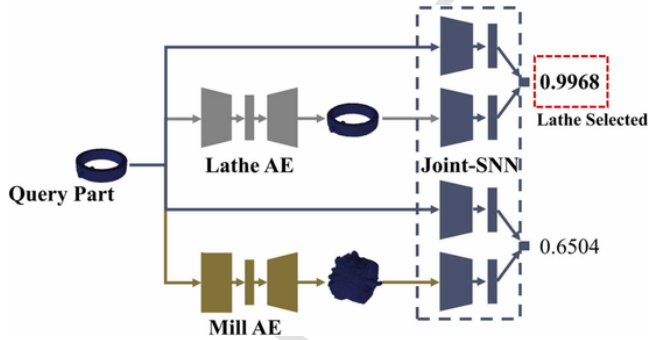


Fig. 11. Example of automated process selection using the AE and SNN models.

ing the workflow introduced in Fig. 2, the query part was input to the AE-SNN to generate the closest matching manufacturable part, and a corresponding similarity score was output to determine manufacturability of the query part. When the query part failed to achieve a certain level of similarity to the output, it was deemed unmanufacturable. In the validation experiment, the AE-SNN was tested using both manufacturable and unmanufacturable parts. The model should accurately classify the manufacturable parts testing data shown in Fig. 6(a) as manufacturable, and the unmanufacturable parts testing data shown in Fig. 6(b) as unmanufacturable. The ROC curve and the confusion matrix for manufacturability analysis using the AE-SNN and Sigmoid classifier models are shown in Fig. 10. The AE-SNN model trained in this work was able to accurately determine manufacturability of the query part drawn from the testing datasets with an accuracy of 100 % and an AUC of 1. On the other hand, the baseline Sigmoid Classifier model under-

performed due to lack of knowledge of unmanufacturable parts data and resulted in an AUC of only 0.25. Furthermore, the Sigmoid classifier requires a 100 % false positive rate to achieve 100 % true positive, indicating that it is unable to analyze the manufacturability of a query part.

It is clear from these results that the baseline discriminative model is not suitable for identifying unmanufacturable parts, since the model is not trained on unmanufacturable parts. This limitation is not applicable to the AE-SNN modeling approach presented in this work. The results indicate a clear advantage of the AE-SNN model over the discriminative model in conducting automated manufacturability analysis in the absence of unmanufacturable parts training data – a situation that is quite common in practice.

4.2. Automated process selection

The second validation experiment aims to evaluate automated process selection capability of the proposed models. In our prior work [36], we have demonstrated that generative 3D-VAE-GAN based models of the shape transformation capability of machining processes can enable a process planner to select a suitable process capable of generating the query part shape. This was enabled through manual (i.e., visual) evaluation of the similarity between the input part shape and the output shape obtained from the generative model.

In the current paper, the similarity score produced by the AE-SNN model is used as the metric for automated process selection. As shown in Fig. 11, when a query part is presented, the AE-only models output the closest matching parts from their respective latent probability distributions, and the SNN component of the joint AE-SNN model outputs simi-

Ground Truth Class	Input	Lathe AE-SNN Output	Mill AE-SNN Output
Lathe Bearing		 0.9912	 0.9339
Lathe Bushing		 0.9965	 0.8776
Lathe Headless Screw		 0.9372	 0.7739
Lathe Rotary Shaft		 0.9965	 0.9162
Lathe Shaft Collar		 0.9958	 0.7153
Lathe Socket Head Screw		 0.9982	 0.8549
Mill Machine Key		 0.9078	 0.9975
Mill Miter Gear		 0.9697	 0.9975
Mill Rectangular Gear Rack		 0.9717	 0.9956

Fig. 12. Example validation outputs from the AE-SNN model for selecting between lathe and milled parts.

larity scores corresponding to known manufacturing processes and automatically selects the process with the highest similarity score. The automated process selection was iterated over the testing dataset, which consisted of 435 lathe-based parts and 32 milled parts. Fig. 12 shows examples of the outputs and the similarity scores obtained from the AE-SNN model. The confusion matrices for the AE-SNN model and the baseline Sigmoid classifier are shown in Fig. 13. The proposed AE-SNN model for automated process selection achieved 87 % accuracy for lathe-based parts, and 91 % accuracy for milled parts, with a class-average accuracy of 89 %, which is comparable to the 94 % class-average accuracy of the baseline Sigmoid classifier model. However, it is important to note that the classification result obtained from the baseline Sigmoid classifier model disregards manufacturability of the query part, which is a clear disadvantage of the discriminative model in an actual manufacturing setting. The proposed generative model, however, ensures that the query part is first manufacturable before automatically selecting a manufacturing process/operation capable of producing the query part.

It is noted that purely associating the shape transformation capability of a manufacturing process with the shape information of a query part is not generally sufficient to determine whether an existing manufacturing resource is available to achieve the desired specification and produce the query part. However, as key elements in manufacturing specification are numerical in nature, factors such as dimensional tolerance can be adopted as constraints after a process is selected based on the shape transformation capabilities of machining processes. For example, if the available milling machine and cutting tool combination is unable to produce parts with tolerance below the maximum value spec-

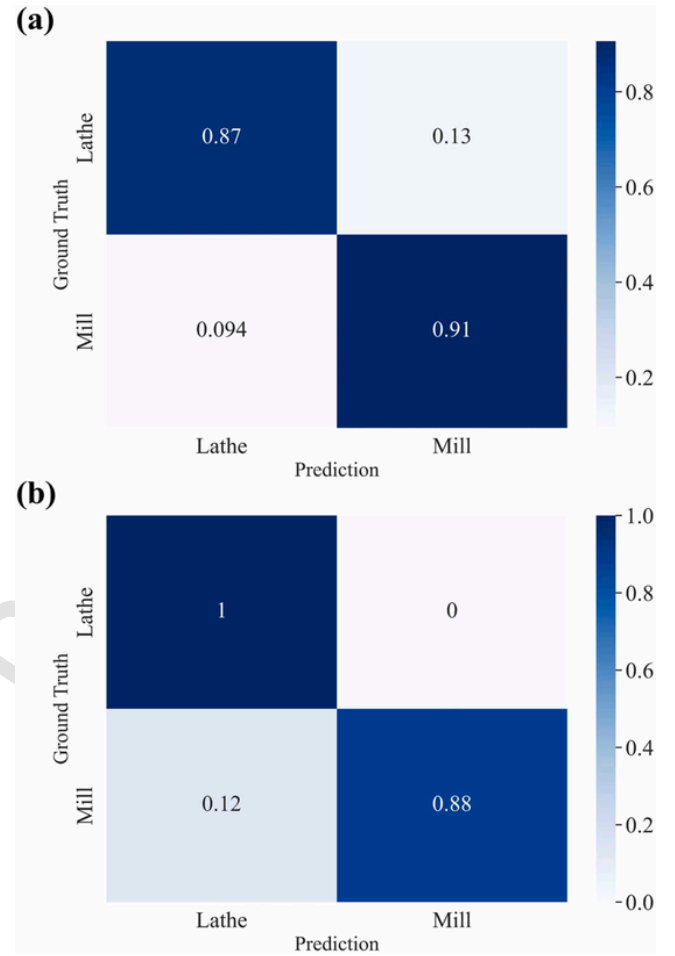


Fig. 13. Confusion matrix for automated process selection for lathe and mill parts (a) AE-SNN, and (b) Sigmoid classifier.

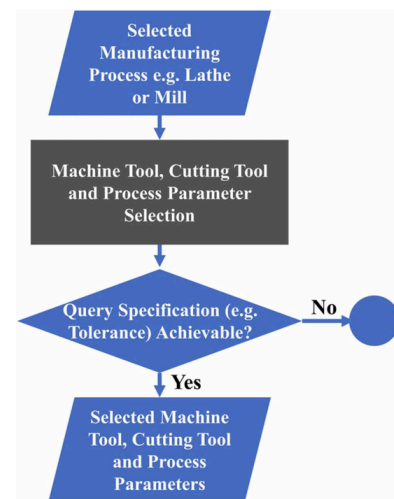


Fig. 14. Envisioned workflow for machine tool, cutting tool, and process parameter selections using query part specification after process selection using methods proposed in this work.

ified a miter gear's shaft deviation tolerance, then the existing milling machine and cutting tool should not be selected to produce the miter gear; conversely, if some milling machine and cutting tool combinations can achieve the desired tolerance, such combinations should be

selected. An envisioned workflow for micro-process planning as possible extension to the work presented in this paper is shown in Fig. 14.

5. Conclusion

In this paper we proposed an AE-SNN deep generative metric learning approach to quantify the shape transformation capabilities of representative machining processes/operations. By modeling the shape transformation capabilities of lathe-based and milled parts using AE-SNN models, we demonstrated that (1) the proposed deep metric learning approach enables both automated manufacturability analysis and automated process selection, (2) the AE-SNN model outperforms the baseline discriminative model in manufacturability analysis with an accuracy of 100 % and a AUC of 1, and (3) the AE-SNN model enables automated process selection with class-average accuracy of 89 %, which is comparable to the baseline discriminative (Sigmoid classifier) model.

There are of course limitations of the models presented here, which will be addressed in future work. One challenge with machine learning-based models in general is the limitation induced by the finite number of training data, which makes it difficult to identify the correct process capability boundary. We acknowledge that the performance of the method proposed here is largely dependent on the quality of the training dataset. With increasing number and diversity of training data, however, the process capability boundary should approach the ground truth. Generalization of the proposed method to out-of-distribution datasets is also not trivial. Since industrial parts typically include many geometric features, a complex part must first be segmented to enable the implementation of the process capability models. Our ongoing efforts are focused on developing and demonstrating this capability. Note that while the proposed AE-SNN model is a multi-tasking deep learning model, a longer training time is required compared to the baseline discriminative model. Optimization of the training process and the model architecture can potentially reduce the training time and improve usability of the trained models in an industrial setting. In addition, the proposed approach should also incorporate material and part quality data, among other data relevant to decision-making, which are critical for manufacturability analysis and process selection. This is also an area of on-going research, which we will report in future work.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was funded by a National Science Foundation EAGER grant (Award # 2113672).

References

- Halevi G, Weill R. Principles of process planning: a logical approach. Springer Science & Business Media.; 1994.
- Trstenjak M, Cosic P. Process planning in Industry 4.0 environment. *Procedia Manuf* 2017;vol. 11:1744–50.
- Al-wswasi M, Ivanov A, Makatsoris H. A survey on smart automated computer-aided process planning (ACAPP) techniques. *Int J Adv Manuf Technol* 2018;vol. 97 (1):809–32.
- Altin L, Zhang H. Computer aided process planning: the state-of-the-art survey. *Int J Prod Res* 1989;vol. 27(4):553–85.
- Chang T.-C, Wysk R.A, Wang H.-P. Computer-aided manufacturing. Prentice-Hall, Inc.; 1991.
- Yusof Y, Latif K. Survey on computer-aided process planning. *Int J Adv Manuf Technol* 2014;vol. 75(1–4):77–89.
- Xu X, Wang L, Newman S.T. Computer-aided process planning—A critical review of recent developments and future trends. *Int J Comput Integr Manuf* 2011;vol. 24 (1):1–31.
- Giachetti R.E. A decision support system for material and manufacturing process selection. *J Intell Manuf* 1998;vol. 9(3):265–76.
- Yu J.-C, Krizan S, Ishii K. Computer-aided design for manufacturing process selection. *J Intell Manuf* 1993;vol. 4(3):199–208.
- Hayes C, Wright P. Automating process planning: using feature interactions to guide search. *J Manuf Syst* 1989;vol. 8(1):1–15.
- Sormaz D.N, Khoshnevis B. Modeling of manufacturing feature interactions for automated process planning. *J Manuf Syst* 2000;vol. 19(1):28–45.
- Turley S.P, et al. Automated process planning and CNC-code generation. IIE Annual Conference. Proceedings. Institute of Industrial and Systems Engineers (IIE); 2014. p. 2138.
- Zhou G, Yang X, Zhang C, Li Z, Xiao Z. Deep learning enabled cutting tool selection for special-shaped machining features of complex products. *Adv Eng Softw* 2019;vol. 133:1–11.
- Prabhu B, Pande S. Intelligent interpretation of CADD drawings. *Comput Graph* 1999;vol. 23(1):25–44.
- Chuang S, Henderson M.R. Three-dimensional shape pattern recognition using vertex classification and vertex-edge graphs. *Comput-Aided Des* 1990;vol. 22(6): 377–87.
- Verma A, Rajotia S. A hint-based machining feature recognition system for 2.5 D parts. *Int J Prod Res* 2008;vol. 46(6):1515–37.
- Venuvinod P.K, Wong S. A graph-based expert system approach to geometric feature recognition. *J Intell Manuf* 1995;vol. 6(3):155–62.
- Korosec M, Balic J, Kopac J. Neural network based manufacturability evaluation of free form machining. *Int J Mach Tools Manuf* 2005;vol. 45(1):13–20.
- Verma A.K, Rajotia S. A review of machining feature recognition methodologies. *Int J Comput Integr Manuf* 2010;vol. 23(4):353–68.
- Zhang Z, Jaiswal P, Rai R. FeatureNet: Machining feature recognition based on 3d convolutional neural network. *Comput-Aided Des* 2018;vol. 101:12–22.
- Ghadai S, Balu A, Sarkar S, Krishnamurthy A. Learning localized features in 3D CAD models for manufacturability analysis of drilled holes. *Comput Aided Geom Des* 2018;vol. 62:263–75.
- Peddiredy D, et al. Identifying manufacturability and machining processes using deep 3D convolutional networks. *J Manuf Process* 2021;vol. 64:1336–48.
- Fu X, Peddiredy D, Aggarwal V, Jun M.B.-G. Improved dixel representation: a 3D CNN geometry descriptor for manufacturing CAD. *IEEE Trans Ind Inform* 2021.
- Wang Z, Rosen D. Manufacturing process classification based on heat kernel signature and convolutional neural networks. *J Intell Manuf* 2022;1–23.
- Ning F, Shi Y, Cai M, Xu W. Part machining feature recognition based on a deep learning method. *J Intell Manuf* 2021;1–13.
- De S, Bermudez-Edo M, Xu H, Cai Z. Deep generative models in the industrial internet of things: a survey. *IEEE Trans Ind Inform* 2022.
- Jiang W, Hong Y, Zhou B, He X, Cheng C. A GAN-based anomaly detection approach for imbalanced industrial time series. *IEEE Access* 2019;vol. 7: 143608–19.
- Belenko V, Chernenko V, Kalinin M, Krundyshev V. Evaluation of GAN applicability for intrusion detection in self-organizing networks of cyber physical systems. 2018 International Russian Automation Conference (RusAutoCon). IEEE; 2018. p. 1–7.
- Hassan M.M, Huda S, Sharmeen S, Abawajy J, Fortino G. An adaptive trust boundary protection for IIoT networks using deep-learning feature-extraction-based semisupervised model. *IEEE Trans Ind Inform* 2020;vol. 17(4):2860–70.
- Wang Y, Perry M, Whitlock D, Sutherland J.W. Detecting anomalies in time series data from a manufacturing system using recurrent neural networks. *J Manuf Syst* 2020.
- Yan S, Yan X. Quality-relevant fault detection based on adversarial learning and distinguished contribution of latent variables to quality. *J Manuf Syst* 2021;vol. 61: 536–45.
- Oh S, Jung Y, Kim S, Lee I, Kang N. Deep generative design: Integration of topology optimization and generative models. *J Mech Des* 2019;vol. 141(11).
- S. Banga, H. Gehani, S. Bhilare, S. Patel, L. Kara, 3d topology optimization using convolutional neural networks, arXiv preprint arXiv:1808.07440, 2018.
- Greminger M. Generative adversarial networks with synthetic training data for enforcing manufacturing constraints on topology optimization. p. V11AT11A005. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. vol. 84003. American Society of Mechanical Engineers; 2020.
- Hertlein N, Buskohl P.R, Gillman A, Vemaganti K, Anand S. Generative adversarial network for early-stage design flexibility in topology optimization for additive manufacturing. *J Manuf Syst* 2021;vol. 59:675–85.
- Yan X, Melkote S. Generative modeling of the shape transformation capability of machining processes. *Manuf Lett* 2022;vol. 33:794–801.
- Kriegel H.-P, Kroger P, Mashaef Z, Pfeifle M, Potke M, Seidl T. Effective similarity search on voxelized CAD objects. Eighth International Conference on Database Systems for Advanced Applications, 2003. (DASFAA 2003). Proceedings. IEEE; 2003. p. 27–36.
- Kaya M, Bilge H.S. Deep metric learning: a survey. *Symmetry* 2019;vol. 11(9): 1066.
- Koch G, Zemel R, Salakhutdinov R. Siamese neural networks for one-shot image recognition. ICML Deep Learn Workshop 2015;vol. 2:0.
- He H, Yang Y, Pan Y. Machine learning for continuous liquid interface production: Printing speed modelling. *J Manuf Syst* 2019;vol. 50:236–46.
- Chicco D. Siamese neural networks: an overview. *Artif Neural Netw* 2021;73–94.
- J. Bjorck, C. Gomes, B. Selman, K.Q. Weinberger, Understanding batch normalization, arXiv preprint arXiv:1806.02375, 2018.
- Xu B, Wang N, Chen T, Li M. Empirical evaluation of rectified activations in convolutional network. arXiv Prepr arXiv:1505.00853 2015.
- B. Starly, A. Bharadwaj, A. Angrish, FabWave CAD Repository Categorized Part

- Classes, ed: DOI, 2019.
- [45] P. Min, "binvox," ed, 2004–2021.
- [46] Paszke A, et al. Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst* 2019;vol. 32:8026–37.
- [47] Kingma D.P, Ba J. Adam: a method for stochastic optimization. *arXiv Prepr arXiv: 1412.6980* 2014.

CORRECTED PROOF