



### Shijie Bian

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330;  
Autodesk Research,  
Autodesk Inc.,  
San Francisco, CA 94111;  
Language Technologies Institute,  
School of Computer Science,  
Carnegie Mellon University,  
Pittsburgh, PA 15213  
e-mail: shijiebian@cmu.edu

### Daniele Grandi

Autodesk Research,  
Autodesk Inc.,  
San Francisco, CA 94111  
e-mail: Daniele.Grandi@autodesk.com

### Tianyang Liu

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330  
e-mail: tianyang.liu@csun.edu

### Pradeep Kumar Jayaraman

Autodesk Research,  
Autodesk Inc.,  
661 University Ave,  
Toronto, ON M5G 1M1, Canada  
e-mail: pradeep.kumar.jayaraman@autodesk.com

### Karl Willis

Autodesk Research,  
Autodesk Inc.,  
San Francisco, CA 94111  
e-mail: karl.willis@autodesk.com

### Elliot Sadler

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330  
e-mail: elliot.sadler.242@my.csun.edu

### Bodia Borijin

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330;  
Department of Structural Engineering,  
University of California San Diego,  
La Jolla, CA 92093  
e-mail: bborijin@ucsd.edu

# HG-CAD: Hierarchical Graph Learning for Material Prediction and Recommendation in Computer-Aided Design

*To support intelligent computer-aided design (CAD), we introduce a machine learning architecture, namely HG-CAD, that recommends assembly body material through joint learning of body- and assembly-level features using a hierarchical graph representation. Specifically, we formulate the material prediction and recommendation process as a node-level classification task over a novel hierarchical graph representation of CAD models, with a low-level graph capturing the body geometry, a high-level graph representing the assembly topology, and a batch-level mask randomization enabling contextual awareness. This enables our network to aggregate geometric and topological features from both the body and assembly levels, leading to competitive performance. Qualitative and quantitative evaluation of the proposed architecture on the Fusion 360 Gallery Assembly Dataset demonstrates the feasibility of our approach, outperforming selected computer vision and human baselines while showing promise in application scenarios. The proposed HG-CAD architecture that unifies the processing, encoding, and joint learning of multi-modal CAD features indicates the potential to serve as a recommendation system for design automation and a baseline for future work. [DOI: 10.1115/1.4063226]*

*Keywords: hierarchical graph learning, graph neural network, computer-aided design, material recommendation, artificial intelligence, graphics processing unit (GPU), computing for design and manufacturing*

<sup>1</sup>Corresponding author.

Manuscript received December 24, 2022; final manuscript received July 25, 2023;  
published online October 4, 2023. Assoc. Editor: Stephen Canfield.

## Thomas Lu

Jet Propulsion Laboratory,  
California Institute of Technology,  
Pasadena, CA 91109  
e-mail: Thomas.t.lu@jpl.nasa.gov

## Richard Otis

Jet Propulsion Laboratory,  
California Institute of Technology,  
Pasadena, CA 91109  
e-mail: richard.otis@jpl.nasa.gov

## Nhut Ho

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330  
e-mail: nhut.ho.51@csun.edu

## Bingbing Li<sup>1</sup>

Autonomy Research Center for STEAHM (ARCS),  
California State University Northridge,  
Northridge, CA 91330  
e-mail: bingbing.li@csun.edu

## 1 Introduction

As a critical aspect in design automation and mechanical engineering, appropriate material selection is a demanding task that requires a devotion of time and expertise through joint analysis of performance, manufacturability, and sustainability [1,2]. Given this challenge, the integration of intelligent tools, such as the application of emerging machine learning algorithms to existing engineering systems, could be explored to assist designers with varying expertise by providing material selection recommendations learned from prior designs in an automated manner [3]. For example, intelligent agents trained on knowledge graphs generated from large-scale design data could help facilitate material selection processes that require extensive evaluation of trade-offs between diverse material candidates. However, there is an opportunity to expand on prior work toward later phases of the design process by extracting design knowledge from computer-aided design (CAD) models to complement existing semantic networks.

In the design domain, CAD tools are used to digitally create 3D models of physical objects and represent various design aspects, including the geometries, topologies, dimensions, tolerances, degrees-of-freedom, material information, and relative motions of components [4]. CAD tools consider material properties and interactivity characteristics between different materials, typically used for simulation to optimize the design or for rendering workflows. Both activities help designers assess and visualize trade-offs between different materials and identify the best materials for the design. Most recently, several large datasets of curated CAD models have been made publicly accessible that support machine learning methods for various data-driven design applications [5–7]. For creating representations of CAD models compatible with machine learning methods, graph data structures have been leveraged to represent design data and to capture various relationships, including semantic relationships between engineering and design concepts [8], joint relationships between parts in assemblies [9,10], and relationships between entities in boundary-representations (BREPs) [10–12]. Simultaneously in product design, efforts to consolidate design knowledge in knowledge graphs have resulted in robust graph representations of domain-specific semantic relationships, which have proven useful for concept generation and evaluation [13–15].

In this paper, we present HG-CAD (Fig. 1), a learning-based approach to predict the material of each part in the assembly. Despite the abundance of 2D drawings and 3D CAD design repositories, the automated prediction of part materials in mechanical design remains challenging due to the ambiguity between the multimodal design knowledge and their relationships to material selection. As illustrated in Fig. 2, the geometry and ground-truth materials of bodies might not have a one-to-one mapping. For example, a screw can be either ferrous or non-ferrous metal, depending on various design requirements such as cost, weight, corrosion resistance, adjacent parts' material, the assembly it belongs to, and its function. This dependency on context introduces additional complexity for existing classification methods that rely on visual recognition. Therefore, their structural and contextual information must be considered to facilitate the accurate classification of bodies within assemblies. Leveraging the expressive power of graphs, we propose a graph neural network (GNN)-based architecture in which material prediction is posed as a node prediction task. Motivated by the importance of material selection to support design automation, this unified architecture is expected to aid designers in selecting appropriate materials by providing part-level material suggestions given a product assembly. We validate the method through quantitative and qualitative evaluation against computer vision and human baselines using the Fusion 360 Gallery Assembly Dataset [9].

To summarize, our work incorporates the following stages and contributions to the areas of design automation and engineering:

- (1) We study the material prediction task for design automation and devise a novel hierarchical graph representation of CAD assemblies to capture geometric and topological information from body and assembly levels.
- (2) We propose HG-CAD, which leverages the hierarchical graph representation of CAD assemblies for automatic prediction and recommendation of materials in unseen assemblies using graph neural networks while preserving contextual awareness using batch-level mask randomization.
- (3) We evaluate the effectiveness of the proposed architecture by conducting quantitative and qualitative experiments, demonstrating competitive performance as compared to

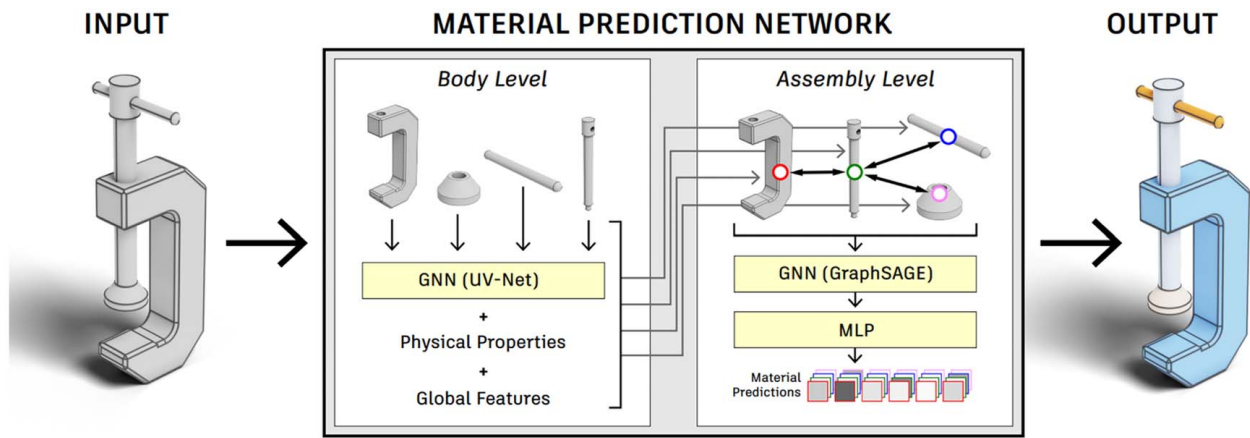


Fig. 1 Proposed method for predicting materials of assembly bodies using hierarchical graph representation and learning

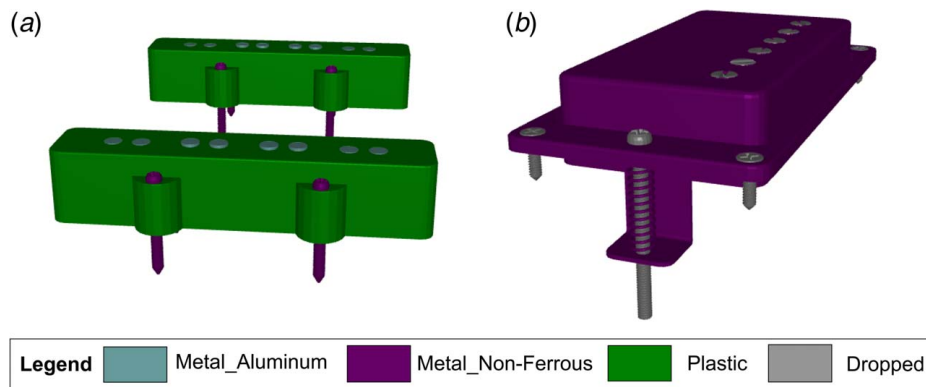


Fig. 2 Two sample assemblies where the color of the assembly body represents the material category. The rounded head screws in assembly (a) and assembly (b) are made of different materials, despite their geometric similarity.

state-of-the-art learning-based models while providing insights and potential applications regarding strengths and limitations through a comparison with a human baseline.

- (4) We provide open-source code and documentation for the proposed HG-CAD architecture (with data extraction, processing, baseline model variations, hierarchical graph construction, and learning toolkit)<sup>2</sup> for reproducibility and further research while being fully compatible with the scalable Fusion 360 Gallery Assembly Dataset [9].

## 2 Related Work

We review prior work on material selection, machine learning algorithms based on deep neural networks, and representation methods tailored to CAD design automation. Reflecting on these past work, we provide deeper insights into the vision and motivation of the proposed work.

**2.1 Material Selection in Product Design.** In product design, material selection can be broken down into a general five-step procedure: (1) establishing design requirements, (2) screening materials, (3) ranking materials, (4) researching material candidates, and (5) applying constraints to the selection process [2]. Performance indices and material property charts, called Ashby diagrams, are often used to visualize, filter, and cluster materials [2,16]. Commercial tools such as Granta CES Edupack allow designers to compare

thousands of materials based on user-input design requirements and constraints [17]. Product designers aim to meet customer needs and technical requirements, but the wide range of consumer products, manufacturing methods, and supply chain availability complicate the material selection process [18]. Accordingly, material aspects such as quality, cost, and function must be considered during product design as they directly contribute to the product's success and its economic, environmental, and social impacts [19–21]. Material selection is further complicated when working with complex assemblies, as the material of individual parts also affects the assembly integration process [22].

Prior work on automated material selection focuses on specific classes of objects (e.g., nozzles and beams) [23–26] and specific design functions (e.g., heat transfer and storage) [27] and is often treated as an optimization problem where the best material is selected based on some required performance criteria [21,28,29]. Despite a few approaches being proposed to leverage neural networks (NNs) for material screening, they are limited in that they do not rank the selected materials [30]. Zhou et al. combine a two-layer NN with a genetic algorithm for selecting appropriate sustainable materials and validate their model on the design of a drink container [31]. Chandrasekhar et al. [26] leverage a variational autoencoder to project a discrete material database onto a differentiable latent space and couple it with a geometry encoder NN to simultaneously optimize the geometry and the material of a beam structure. Our method differs in that it only requires design information commonly documented in CAD, is agnostic to the manufacturing method and class of objects, and does not aim to find the most optimal material but suggests top-ranked appropriate materials to

<sup>2</sup><https://github.com/BrandonBian/hg-cad>

the designer during the design process, thereby supporting user creativity and design flexibility.

**2.2 Convolutional and Graph Neural Networks.** Convolutional neural networks (CNNs) are a class of artificial neural networks predominantly used for feature representation and learning. Specifically, CNNs apply filters via sliding a kernel over the multi-dimensional input embeddings to perform convolution and pooling processes that extract, summarize, and represent the presence of detected features, which can be subsequently passed through a feed-forward neural network for representation learning tailored to tasks such as classification through activation. Due to the flexibility of the kernel dimensions, CNNs can effectively extract features of various scales of details that can be pooled into a unified representation and are thus most dominantly used for visual representation in computer vision [32,33].

Like CNNs, GNNs are a category of deep neural networks designed to perform learning and inference on data described by graphs and distributed in a non-Euclidean space. Message passing for representation learning in GNNs mainly consists of two main steps: aggregation, in which we collect features from topological neighboring nodes and edges; combination, in which we integrate the aggregated features into a single-node representation, later pooled to form graph representations. Specifically, GNNs learn representations over order-invariant data structured as graphs and often of variable sizes through an iterative process of transferring, transforming, and aggregating the representations with topological awareness [34–39]. The learned representations are combined into a graph-level representation, which can be used to perform tasks such as node, edge, and graph-level classifications.

In the proposed work, we leverage both types of neural networks to enable adequate representation and learning of CAD through joint exploitation of geometrical and topological features. In summary, the proposed architecture represents both CAD assemblies and bodies as hierarchical graphs, where the bodies are structured as graphs within assembly graphs and the assemblies are structured as super-nodes consisting of sets of body graphs, similar to the architecture proposed by Xing et al. [40]. CNNs are utilized for body-level geometrical representation extraction and encoding, whereas GNNs are adopted to perform topological representation learning with structural and contextual feature awareness. The proposed work's combination of both neural networks in an end-to-end paradigm allows for joint geometrical and topological learning, thus exploiting the utilization of multi-modal CAD features to the full extent, allowing for classification and predictions with effectiveness and efficiency exceeding that of mono-modal representation learning.

**2.3 Graph Representation of Computed-Aided Design Models.** The design process is iterative and generates large amounts of data that can be organized and parsed for additional information that may be used to improve the design [41]. This information, collected from all aspects of the product life-cycle, is multi-modal and can be in the form of semantic names, customer requirements, 3D geometry, material properties, manufacturing tolerances, cost information, etc. These data may then be used to modify the design itself, enabling some design process automation by learning from prior examples [42,43]. Prior work looked at organizing and learning from design knowledge acquired from sources such as taxonomy-based design repositories [44,45], product tear-downs [46], patent data [8], and geometry-based design repositories [47–49]. When working with geometric data, graphs have been leveraged to represent BREP geometries with goals ranging from representing complex relations to solving design problems [11,50–52]. In particular, UV-Net [12] is a novel neural network architecture that can be leveraged for classification and segmentation tasks on B-rep data from 3D CAD models. UV-Net represents each assembly body as a face-adjacency graph, with face and edge features represented as a structured grid of UV-grid and U-grid of points, respectively.

Convolutional neural networks are applied on the face and edge features that are further message passed using a graph neural network. The highly expressive nature of graphs in discrete encoding can also be applied to enable the more sophisticated representation and learning of multi-modal data. For example, Jones et al. [53] proposed a structured BREP graph convolution network that utilizes structured graph representation to encode heterogeneous BREP information, which prompted the creation of an assembly modeling tool for automatic mating of assemblies by effectively capturing the topological relations of parts. Similarly, Pfaff et al. [54] applied graph neural networks to mesh-based simulation, with an encoder that transforms mesh into graphs with the addition of edges, a processor that performs convolutional message passing, and a decoder that extracts signal for updates.

Our proposed work expands on prior literature by augmenting CAD representations through a novel hierarchical graph representation and learning architecture, effectively capturing the geometrical and topological features on both body and assembly levels while preserving contextual and structural awareness through GNN learning with node masking for CAD material prediction and recommendation.

### 3 Methodology

This section introduces the proposed HG-CAD architecture and the motivation behind the preprocessing steps for the CAD dataset. Next, we formulate the proposed hierarchical graph representation of CAD models and the detailed methodology for classifying and predicting assembly body materials using joint geometrical and topological learning with batch-level mask randomization.

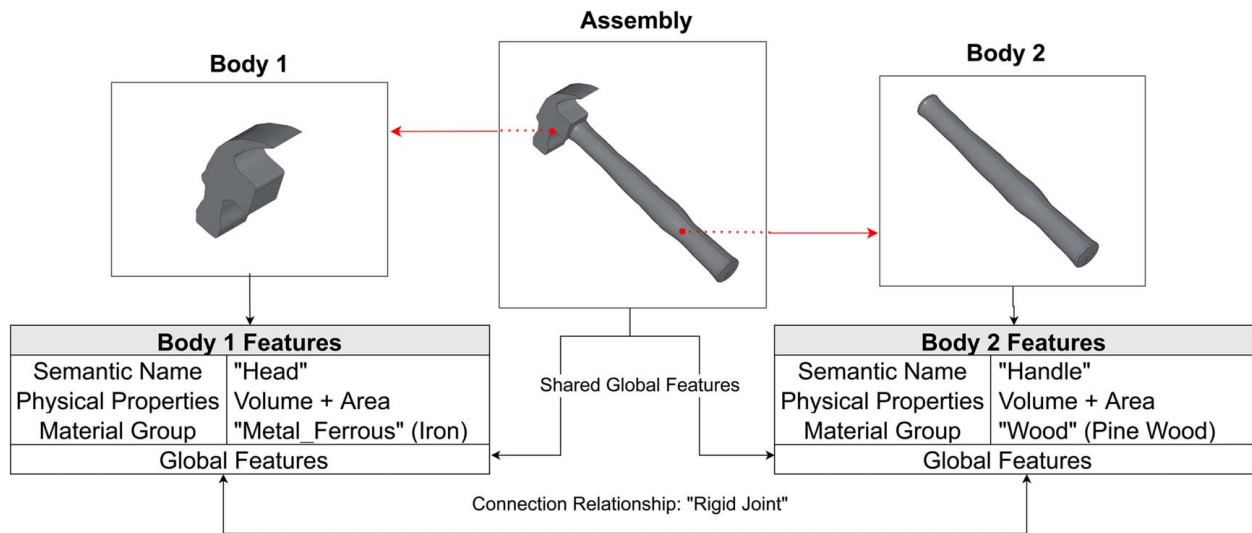
**3.1 Architecture Overview.** As illustrated in Fig. 1, the proposed material prediction network consists of two primary hierarchies: (1) the body-level module that extracts and processes the CAD body features, represents them in terms of body graphs to capture topology, and encodes the geometrical and physical features using different tools and transformations; (2) the assembly-level module that provides a graph representation of the entire assembly, passes the topological information and the body-level features collected from the previous module through a graph neural network for embedding generation, and using a densely connected neural network for representation learning and prediction.

**3.2 Dataset and Preprocessing.** The proposed architecture is compatible with the publicly available Fusion 360 Gallery Assembly Dataset [9]. As illustrated in Fig. 3, each *assembly* in the dataset has assembly-level feature properties, which are shared across numerous *bodies* connected via assembly relationships. Each assembly body has a set of body-level design features, and the bodies are manually organized by users in a hierarchy of *occurrences*, which are the building blocks that make up the *assemblies*. The dataset contains 154,468 bodies that are grouped into 8251 assemblies, created across different industries and design categories and with various levels of detail. This dataset was selected because of its scalable size, abundant diversity of designs and multi-modal features, ease of processing, and presence of body-level material labels. Due to the complexity and diversity of the dataset, preprocessing and feature encoding steps are performed to target the assemblies and bodies of interest and encode the extracted multi-modal raw features into comprehensible formats in the learning architecture. The preprocessing steps on the dataset are as follows.

*Data filtering.* We drop assemblies that are entirely labeled with the default material (*Steel*), resulting in 6336 assemblies.

*Material label transformation.* Each assembly body within the assembly dataset is defined with two material labels: a *physical material label* that defines the physical and mechanical properties





**Fig. 3 An example assembly of two bodies with corresponding features and connection relationships**

of the assembly body for simulation and an *appearance material label* used for displaying and rendering the object. Despite the different labels serving distinct purposes, it may be assumed that the designer intends to have each assembly body defined by one material. To reflect this, a transformation process is performed in which the non-default appearance label is selected as the material ground truth to replace the default physical material label, if applicable. This is performed to improve the quality and abundance of ground-truth labels while attempting to preserve the original design intent.

**Material label grouping and dropping.** Due to the diversity, skewness, and sparsity in the distribution of user-defined body-level materials, a manual regrouping step is taken to reorganize the ground-truth labels. Specifically, the detailed material labels, as transformed from the previous step, are mapped to their corresponding generalized material categories as defined in the Autodesk Fusion 360 material library, which are then regrouped into eight simplified material groups, as summarized in Table 1, based on a combination of their primary, secondary, and tertiary subcategories. After performing the regrouping of material categories for body-level ground-truth labels, the following two groups are dropped: *Metal\_Ferrous\_Steel*, which contains only the default carbon steel material and is thus less relevant to the material prediction

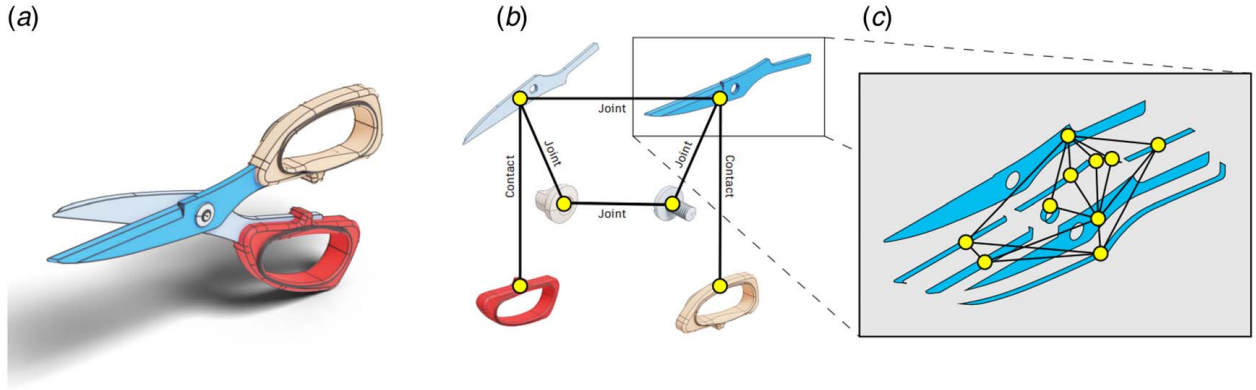
task, and *Paint*, which contains generic and rendering material that is ambiguous and does not reflect meaningful design or physical properties. More specifically, the motivation for dropping the *Metal\_Ferrous\_Steel* label is double-fold: (1) it is configured as the default material label for design bodies in the Autodesk Fusion 360 platform, which makes it irrelevant as they do not contain sufficient user-configured information to benefit the model's training, and (2) it is a dominant label that introduces label imbalance which induces prediction bias. Therefore, the regrouping step of ground-truth labels effectively reduces the complexity and distributional skewness while preserving logical correctness, whereas the dropping of default and paint-related material groups alleviates the confusion incurred during the architecture learning process. In addition, incorporating the final six simplified material groups in the prediction pool enables more flexible and scalable recommendation outputs than solely considering the exact material labels, thus leaving room for the creativity and freedom of designers.

**3.3 Hierarchical Graph Representation.** Despite multi-modal expressive features, the CAD assemblies and their corresponding bodies are difficult to encode and may introduce complications when directly delivered into machine learning architectures. Therefore, we represent CAD assemblies using a graph representation, wherein, body attributes are captured as node features and topology is captured using edges:

**Table 1 Table of the simplified ground-truth material groups**

Simplified material group	Definition	Example(s)	Label count
Metal_Aluminum	Aluminum-based metal	Aluminum alloy	10,606
Metal_Ferrous	Ferrous metal (excluding carbon steel)	Cast iron	7138
Metal_Ferrous_Steel (dropped)	Carbon steel	Carbon steel	39,891
Metal_Non-Ferrous	Non-ferrous metal	Platinum, silver	16,276
Other	Uncategorized material	Glass, fabric	15,028
Paint (dropped)	Generic, rendering, and coating material	Metal flake	13,193
Plastic	Plastic	Thermoplastic	20,063
Wood	Natural and engineered wood	Softwood	9107

**3.3.1 Body-Level Graph Representation.** As illustrated in Fig. 4(c), we define *body graphs* as attributed and directed face-adjacency graphs representing CAD assembly bodies, where the visible parametric surfaces are represented as nodes, and the visible interval of the parametric boundary of faces are represented as edges that connect two faces that are adjacent to each other. For graph feature generation, we follow the methodology as introduced in UV-Net. Specifically, we generate the *body graph node features* by parameterizing assembly body surfaces as sets of 2D features constituted of the absolute 3D normalized coordinates sampled from surface domains with a uniform step size. Similarly, *body graph edge features* are created by parameterizing boundaries as sets of 1D features constituted of absolute point coordinates sampled from its parameter domain with a uniform step size. The node and edge feature matrices generated by a stacking of individual feature vectors, along with the connectivity information of nodes, are organized into deep graph library (DGL) [55] graph object instances and are subsequently delivered into the learning



**Fig. 4** We represent each CAD assembly (a) as a graph with two hierarchical layers. Our top-level assembly graph (b) contains bodies as nodes and connection types as edges. At the next level, body graph (c) contains B-Rep faces as nodes connected by adjacency edges.

architecture along with the assembly-level graphs created according to the subsequent section.

**3.3.2 Assembly-Level Graph Representation.** As illustrated in Figs. 4(a) and 4(b), we define *assembly graphs* as attributed and directed multi-graphs representing entire CAD assemblies, where bodies corresponding to individual parts are encoded as graph nodes. Connections representing assembly relationships between the bodies are encoded as edges between corresponding nodes.

When constructing *assembly graph node features*, we consider the following three assembly body physical properties: body area represented in square meters, body volume represented in cubic meters, and body center of mass represented in the  $x, y, z$  coordinates. All physical properties are generated by Autodesk Fusion 360 and are numerical. Standard scaling transformation is applied to each physical property to generate their corresponding physical feature embedding vectors. We further consider the features defined at the assembly level and are thus shared among the bodies of the same assembly globally. These features include the assembly physical properties and assembly geometric properties, which are integrated from individual assembly body properties, as well as design category, industry, and products that the assembly belongs to. Generally, global features represented as floating-point or integer numbers are preserved and normalized using standard scalar transformation, whereas categorical features represented as character strings are one-hot encoded. When constructing *assembly graph edge features*, we entail the following three predominant types of connection relationships: *contacts* define the relationship between two bodies whose faces are in contact with each other; *joints* define the relationship between two bodies whose relative pose and degrees-of-freedom are constrained; *occurrence relationship* represents the user-assigned relative hierarchy of the bodies, such as multiple bodies sharing the same occurrence, or distributed across a parent-child occurrence relationship. These three connection types, being categorical, are encoded using the one-hot method and serve as the edge features.

The resulting graph is attributed since each node and edge is attached with features that effectively capture the physical and structural information. The graph is directed since each edge contains a source node, a destination node, and a direction that correlates the two nodes. Furthermore, the graph is a multi-graph since it allows the existence of multiple edges between any pair of nodes. Once the assembly graphs are constructed, instances with less than three nodes or less than two edges are considered trivial and are thus discarded. Following the reasoning in Sec. 3.2, the nodes with ground-truth material category labels of `Metal_Ferrous_Steel` and `Paint` are dropped, and the edges connected to them are removed accordingly from the

connectivity matrices. The node and edge feature matrices are generated by stacking individual feature vectors and the shared global features. The feature matrices and the connectivity information of nodes represented in coordinate (COO) format are organized into PyTorch Geometric [56] graph objects and are subsequently delivered into the learning architecture along with the body-level graphs created according to the prior section.

**3.4 Hierarchical Learning Architecture.** Building on the hierarchical graph representation of CAD assemblies, we present the learning architecture of HG-CAD, in which we utilize graph neural networks to generate graph embeddings in both body and assembly hierarchies by passing node-level feature messages through convolution.

We denote a set of *assembly graphs* of interest constructed from Sec. 3.3.2 as  $\mathcal{G} = \{\mathcal{G}_k\}_{k=1}^N$ , with individual graphs as  $\mathcal{G}_k = (\mathcal{V}, \mathcal{E}, G, \mathcal{M}, \mathcal{Y})$ , each of which consists of a set of *body nodes*  $\mathcal{V} = \{v_1, v_2, \dots, v_{|\mathcal{V}|}\}$  (carrying node features  $h_i \in \mathbb{R}^{d_v}$ ), a set of *assembly relationship edges*  $\mathcal{E} = \{e_{ij} = (v_i, v_j)\} \subseteq \mathcal{V} \times \mathcal{V}$  (carrying edge features  $h_{ij} \in \mathbb{R}^{d_e}$ ), a set of *body graph embeddings*  $G = \{g_1, g_2, \dots, g_{|\mathcal{V}|}\}$  corresponding to each body node, a *masking matrix*  $\mathcal{M} = \{\bar{m}_1 | \bar{m}_2 | \dots | \bar{m}_{|\mathcal{V}|}\}$  such that  $\bar{m}_i \in [0, 1]$  ( $i \in 1, \dots, |\mathcal{V}|$ ) representing the validity of body graph embedding, and a set of ground-truth *material category labels*  $\mathcal{Y} = \{y_1, y_2, \dots, y_{|\mathcal{V}|}\}$  corresponding to each node. The topological representation learning process can be formulated as a supervised node classification task, in which a node-level representation  $h_v$  ( $\forall v \in \mathcal{V}$ ) is learned from a combination of the assembly graph topology, assembly graph node and edge features, and the body-level geometric embedding, such that the ground-truth material category label for each body can be predicted.

During the training time, at a certain  $l \in \{1, \dots, L\}$  layer of the graph neural network, message passing is performed between direct neighbors through neighborhood aggregation, where the representation of a certain node  $v_i \in \mathcal{V}$  is iteratively updated by a combination of aggregated neighboring node and edge features, as defined below:

$$h_{v_i}^{(l)} = \phi_c^{(l)} \left( h_{v_i}^{(l-1)}, \bigcup_{v_j \in \mathcal{N}_i} \phi_a^{(l)} \{ h_{v_i}^{(l-1)}, h_{v_j}^{(l-1)}, h_{ij} \} \right) \quad (1)$$

where  $h_{v_i}^{(l)}$  denotes the representation of node  $v_i$  at layer  $l$  of the neural network, and  $\phi_c(\cdot)$  and  $\phi_a(\cdot)$  denote the parametric combination and aggregation functions.  $\mathcal{N}_i$  denotes the topological neighbors of node  $v_i$ , and the representation of node  $v_i$  during the aggregating step contains a concatenated feature of body-level geometrical representation generated from the DGL body graphs  $g_i$  and the corresponding masking matrix for the elimination of invalid

embeddings, as defined below:

$$\mathcal{N}_i = \{v_i \in \mathcal{V} : \exists e_{ij} \in \mathcal{E}\}, \quad h_{v_i}^l = [h_{v_i}^l \| g_{v_i}^l \vec{m}_{v_i}] \quad (2)$$

where the body graph geometrical embedding of a graph node at a given layer  $g_{v_i}^l$  is generated by a summation aggregation of the body graph's node features (2D convoluted from parameterized surface coordinates) and edge features (1D convoluted from parameterized boundary coordinates) using a multi-layer perceptron, following the learning architecture as introduced in UV-Net [12].

GraphSAGE [57] is selected for the assembly-level graph neural network layers, utilizing a mean aggregator. The graph connectivity information, concatenated node, and edge feature matrices are passed into the GNN encoder during the training process. The node and edge embedding is obtained after message passing of each neural network layer, and non-linear activation via leaky rectified linear unit [58] is performed. For generating the final node embedding to perform node classifications on the input graph, a similar procedure as the jumping knowledge networks [59] is performed by aggregating the node embedding of individual GNN layers from the learning architecture through max-pooling. The final embedding is then passed through a multi-layer perceptron with batch normalization layers [60], parametric rectified linear unit [61] for non-linearity activation, and softmax activation for predicting class distributions. Weighted cross-entropy loss is adopted to prevent the neural network from overlooking rare classes. The weights for each class are initialized as inversely proportional to the ground-truth class frequencies.

Considering application scenarios, we propose *batch-level mask randomization*, in which ground-truth material category labels are injected into portions of assembly graph nodes using randomized masking to prompt the neural network learning toward fully exploiting the contextual design knowledge and user-introduced information. Motivated by the fact that designers may have access to material information for completed portions of their design assemblies, we inject ground-truth material category labels as additional node features into all but one of the assembly graph nodes (referred to as *context* nodes) and predict the labels for the one node without the ground-truth information (referred to as *target* nodes). For the single *target* node of the assembly graph, a vector of zeros is attached to the node feature matrix to simulate the masked ground-truth information. Only the ground truth and predictions of the batched assembly graphs' *target* nodes are involved in the loss calculation process during training and validation steps and the accuracy calculation process during testing. During the message passing process within the neural network layers, the *context* nodes pass their embedding through the edges to the *target* nodes, updating their embedding. This prompts the GNN to predict the *target* nodes by learning from both the topology of the graph and the ground-truth labels contained within the neighboring *context* nodes, thereby achieving structural and contextual awareness.

## 4 Experiments

In this section, we analyze and evaluate the performance of our architecture through experiments on the Fusion 360 Gallery

**Table 2 Table of the hyperparameters of the model and baselines**

Model	Epochs	Learning rate	Batch size	Hidden dimension
HG-CAD (ours)	Early stopping of 30	0.001	16	256
UV-Net	350	0.001	128	64
PointNet	Until convergence	0.001	128	256
MVCNN	30	0.0001	8	512

assembly dataset. Specifically, we consider the following three baselines of machine learning classification models: (1) *UV-Net* [12], which exploits geometry as sampled from solid surfaces and edges together with topology structured as graphs; (2) *PointNet* [62], which samples point clouds on mesh surfaces to perform object classification on CAD models; (3) *MVCNN* [63], which uses a CNN architecture that combines 2D geometry obtained from multiple views of a 3D model to perform recognition and inference. We also consider a *human baseline*, for which we ask students from mechanical engineering backgrounds to provide manual body-level predictions on existing design assemblies.

**4.1 General Setup.** To ensure a fair comparison, we split the constructed assembly graphs obtained from Sec. 3.3.2 into the training, validation, and test sets, which remain fixed across all models (but excluding the human baseline). Specifically, we generated the training, validation, and test sets based on the proportions of 64%, 16%, and 20%, respectively, through a random sampling process on the assembly graphs, repeated for a total of three iterations with different random seeds. Furthermore, we provide the configuration of the proposed model and the baselines to ensure reproducibility, as summarized in Table 2 and detailed in the following paragraphs.

*HG-CAD (ours).* The assembly-level graph neural network consists of seven GraphSAGE layers with a hidden dimension of 256. This configuration is determined through an extensive grid search on the number of GNN layers (ranging from 1 to 8) and the size of hidden dimensions (64, 128, 256, and 512). For the training process, we adopted the Adam optimizer [64] with a learning rate of 0.001 and a cosine annealing scheduler [65] with the maximum iteration set to be equivalent to the number of epochs for learning rate tuning. After obtaining the embedding from each layer, a leaky rectified linear unit with a negative slope set to 0.2 is applied for non-linear activation. For the body-level learning architecture, we follow the same setting as that of UV-Net. The architecture is implemented with a combination of PyTorch geometric and PyTorch lightning. Training is performed with an early stopping of 30 epochs, and the model with the least validation loss is selected as the best for testing. To further investigate the effectiveness of the hierarchical graph representation in capturing contextual information, we configured our model to produce a variation in which the randomized node masking is removed.

*UV-Net.* By removing from our model the assembly-level GNN, we are left with only the part-level GNN, which leverages UV-Net to represent the part geometry. We use the official implementation<sup>3</sup> of the UV-Net [12] classification model where the final linear layer was modified to output six classes. Each B-rep body is converted into face-adjacency graphs with 2D and 1D UV-grids as node and edge features, respectively, where we used ten sample points for the *u*- and *v*-directions as in the original implementation. The model was trained with the Adam optimizer [64] for 350 epochs, and the weights of the model corresponding to the minimum validation loss were used in our experiments.

*PointNet.* For each input body-level geometry encoded as triangulated B-Rep representation, 2048 points are randomly sampled. An Adam optimizer with a learning rate of 0.001 and a momentum of 0.9 is used. Sparse categorical cross-entropy is used to calculate loss, and a dropout rate of 0.3 is used on the last fully connected layer of hidden dimension 256 to alleviate overfitting. PointNet is implemented using TensorFlow, and the model is trained until the convergence of validation loss.

*MVCNN.* The MVCNN is trained using a PyTorch implementation and uses the ResNet architecture [66] with a supervised learning regime. A patience factor is used to stop the training process

<sup>3</sup><https://github.com/AutodeskAILab/UV-Net>



after 20 epochs, which increases the validation accuracy, resulting in around 30 training epochs. The models are trained with eight as the batch size, 512 embedding dimensions, 12 views, and  $1 \times 10^{-4}$  learning rate.

**Human baseline.** To compare the performance of the proposed work to that of humans, we asked three Master's students to provide their predictions of material category labels for assembly bodies. Specifically, we randomly sampled 300 assemblies from the entire dataset and dropped the assembly bodies whose ground-truth material groups were *Metal\_Ferrous\_Steel* or *Paint*, simulating the process as described in Sec. 3.2. To ensure fairness of comparison, we sought to reproduce the information provided to the baseline models by presenting the human labelers with an interactive user interface containing the following content: 2D thumbnail images of the assembly body and its corresponding assembly, 3D rendered and interactive display of the assembly geometry with the body portion highlighted, semantic names and physical properties of the body, as well as the assembly-level global features. For each assembly body, the students are asked to classify its material into one of the six material groups listed in Table 1 based on their experience, expertise, and the provided information. Figure 5 shows an example human labeling template.

**4.2 Quantitative Evaluation.** Table 3 summarizes the quantitative results of the experiments compared between the proposed model and the baselines. To evaluate the multi-class classification performance and consider the skewness of ground-truth material group distribution, we record the micro  $F_1$  score calculated by weighting each prediction instance equally. The performance statistics of each model are recorded per iteration and are averaged across the iterations to calculate the mean and standard deviations. For evaluating the human baseline, the human labelers' body-level predictions are concatenated and compared via one-to-one mapping to the corresponding assembly body's ground-truth material group. Due to having only one iteration over the 300 assigned assemblies, the human baseline performance does not have a standard deviation.

From the numerical results, we infer that material prediction in CAD is generally a highly challenging task for both machine learning models and human labelers, due to the highly variational

distribution of body geometry and assembly ground truth. The proposed HG-CAD, when producing predictions of all body materials of an assembly without referencing contextual material information, produced results that are only sparingly above that of the baseline. However, when enabling the joint learning of assembly topology and part geometry with contextual information using node masking, the proposed architecture achieves an average of 0.59 micro  $F_1$  score, which surpasses that of the baseline machine learning models that mainly rely on geometry. This supports our hypothesis that the task of material selection depends on multiple factors other than the geometric features of the design, namely the topology of assembly bodies that enable contextual awareness during hierarchical graph learning. To further investigate the performance discrepancies and the human baseline results, we plot the classification confusion matrices for the models displayed in Fig. 6.

An ideal confusion matrix for a classification model with the best classification should demonstrate clear diagonal patterns with no shades in other regions. From the confusion matrices, we observe that UV-Net and PointNet misclassify most material categories as *Plastic*, as observed by the significant shades in the plastic column. This is because the *Plastic* category is a majority class, with many samples dominating the other classes. The imbalance of label count could have misguided these two models to make predictions biased toward the *Plastic* category. UV-Net demonstrated less confusion and thus better classification performance than PointNet, possibly due to the inclusion of body-level topology with graph representations. MVCNN showed a clearer diagonal pattern, but confusion still persists with the *Metal\_Non-Ferrous* and *Other* groups. The proposed model, deprived of material context information shared during message passing and learned via node masking,

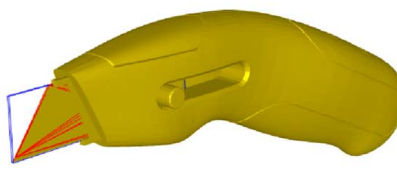
**Table 3 Table of the experimental results for the material prediction task**

	HG-CAD	HG-CAD (no mask)	UV-Net	PointNet	MVCNN	Human
Acc.	$0.59 \pm 0.031$	$0.30 \pm 0.020$	$0.29 \pm 0.012$	$0.26 \pm 0.008$	$0.26 \pm 0.008$	0.35

-----  
Body Material Selection:

Metal_Aluminum	Metal_Ferrous	Metal_Non-Ferrous
Plastic	Wood	Other

-----

<p style="text-align: center;"><b>Assembly</b></p> <ul style="list-style-type: none"> <li>- Product = ['Fusion 360']</li> <li>- Industry = ['Product Design &amp; Manufacturing']</li> <li>- Category = ['Product Design']</li> </ul>	<p style="text-align: center;"><b>Body: "body "   Occ: "blade "</b></p> <ul style="list-style-type: none"> <li>- Body Properties: Area = 16.26   Volume = 0.45</li> <li>- Occ Properties: Area = 16.26   Volume = 0.45</li> <li>- Body Center of Mass = (0.0, -0.05, 3.86)</li> </ul>	
<p style="text-align: center;"><b>Additional Assembly Info</b></p> <ul style="list-style-type: none"> <li>- Physical Properties (Pt.1): # Edges = 954   # Faces = 354   # Loops = 374   # Shells = 6   # Vertices = 624</li> <li>- Physical Properties (Pt.2): Volume = 24.72   Center of Mass = (0.02, -0.92, -2.41)</li> <li>- Community Statistics: # Likes = 0   # Comments = 0   # Views = 347</li> </ul>		

**Fig. 5 An example graphical user interface display for human baseline**



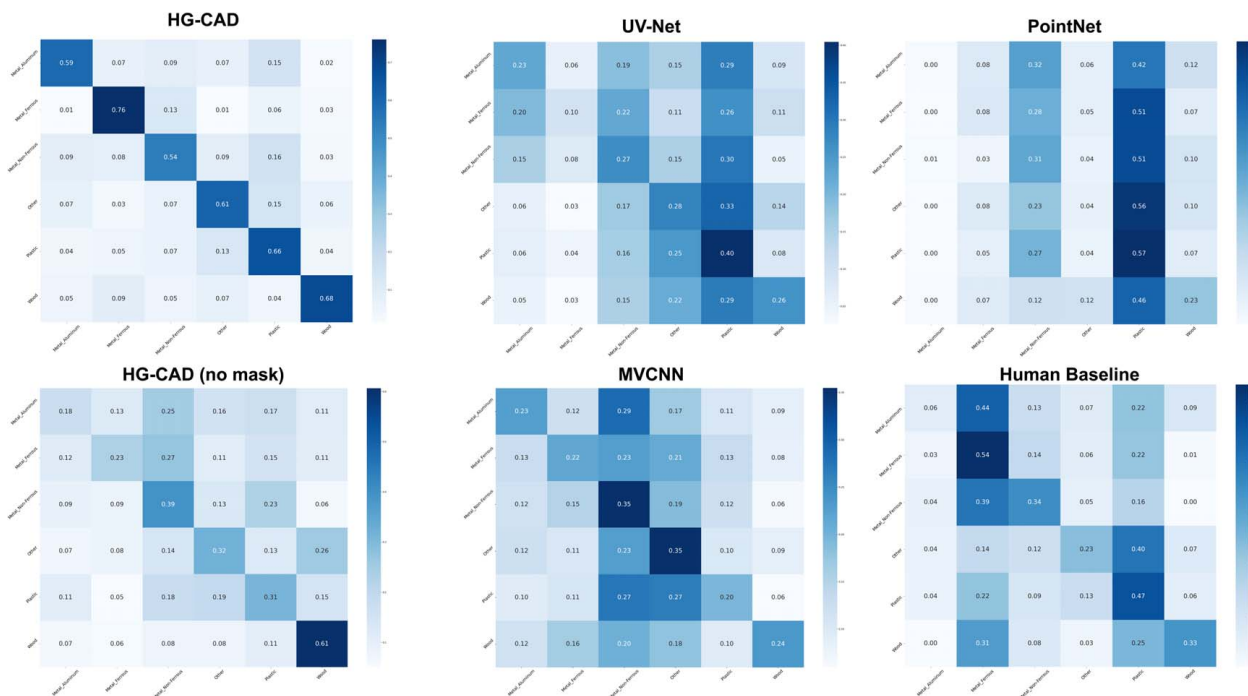


Fig. 6 A comparison of classification confusion matrices for the material prediction task

demonstrated similar confusion as the computer vision baselines, though with a slight alleviation in severity. Significant confusion is observed for the human baseline, where no assembly body was classified as *Metal\_Aluminum*. This may be because *Metal\_Aluminum* is a precise and exclusive class compared to the other five more general and inclusive categories, and human labelers confused aluminum material with a non-ferrous metal material. Compared to the four baselines, HG-CAD’s classification performance is superior, showing a much clearer diagonal pattern and, thus, less confusion during prediction. There is still a slight prediction pattern biased toward the majority class of plastic, but the effect is minimal. These results, together with the observation illustrated in Fig. 2, demonstrate that representation learning on geometry only is insufficient for making body-level material prediction and classification and that the inclusion of topology via graphs as well as neighborhood context via random masking can greatly reduce confusion.

**4.3 Qualitative Evaluation.** To investigate the prediction results in a real-world use case setting and to gain further insights for potential application, we provide a set of rendered assemblies to compare the prediction results of our model and that of the baselines in a qualitative manner. Specifically, we select four assemblies from a union of the test set of the machine learning-based models and that of the human baseline. The chosen assemblies are delivered into the machine learning models pre-trained on the fixed train-test split, and the inference results of body-level material predictions are produced. Subsequently, the 3D models of the chosen assemblies are rendered with colors corresponding to the predicted material categories.

Illustrated in Fig. 7 is the visualized qualitative comparison. The gray color represents the material categories *Metal\_Ferrous\_Steel* and *Paint*, which are dropped and therefore do not participate in the evaluation process. From the visualized comparison, our proposed model can function well when dealing with bodies of similar geometry and symmetrical distribution in space. For example, our model predicted almost perfectly the material category of the pistons in the first sample assembly, whereas the predictions of the other baseline models are rather arbitrary. This demonstrates that our model can effectively exploit the body-level geometrical

information in conjunction with assembly-level topological information through the proposed joint representation learning methodology. For the performance on the latter three assemblies, the baseline models gave predictions inconsistent with the ground truth, often confusing the aluminum metal category (blue) with the non-ferrous metal category (orange). A similar confusion within the metal categories can be observed for the human baseline predictions, where almost all metal occasions are misclassified as *Metal\_Ferrous*. The qualitative evaluation demonstrates the effectiveness of the proposed joint learning of topological and geometrical representation methodology compared to baseline methods that rely on only a single aspect of CAD representation.

## 5 Discussion

### 5.1 Limitations and Improvements

**5.1.1 Data and Model Augmentation.** While running single-node prediction experiments, we observed that the inference confusion decreases with increasing training and validation assemblies. Despite the batch-level randomized node masking process, only one node per assembly graph participates in the loss calculation for training and validation; thus, the support for each class is diminished, which may lead to problems such as overfitting. Therefore, additional steps should be taken to perform data and graph augmentation to alleviate this issue. For collecting additional CAD data with a more varied distribution of features and ground-truth materials, we seek to expand the currently adopted Fusion 360 Gallery Assembly Dataset by inviting more designers to contribute or by using additional publicly available datasets such as ABC and Auto-Mate [6,53]. For graph-level augmentation, existing graphs can be manipulated to generate additional training and validation samples, such as by creating sub-graphs or synthesized graphs from past designs through node dropping, edge dropping, or sub-graph sampling [67]. Existing tools designed to support automated reasoning, such as Open-NARS [68], could also be applied to the machine learning process to support the inference and imputation of rare material classes. Since neural networks require abundant high-quality training samples, incorporating the reasoning system might provide a more balanced approach through deduction,

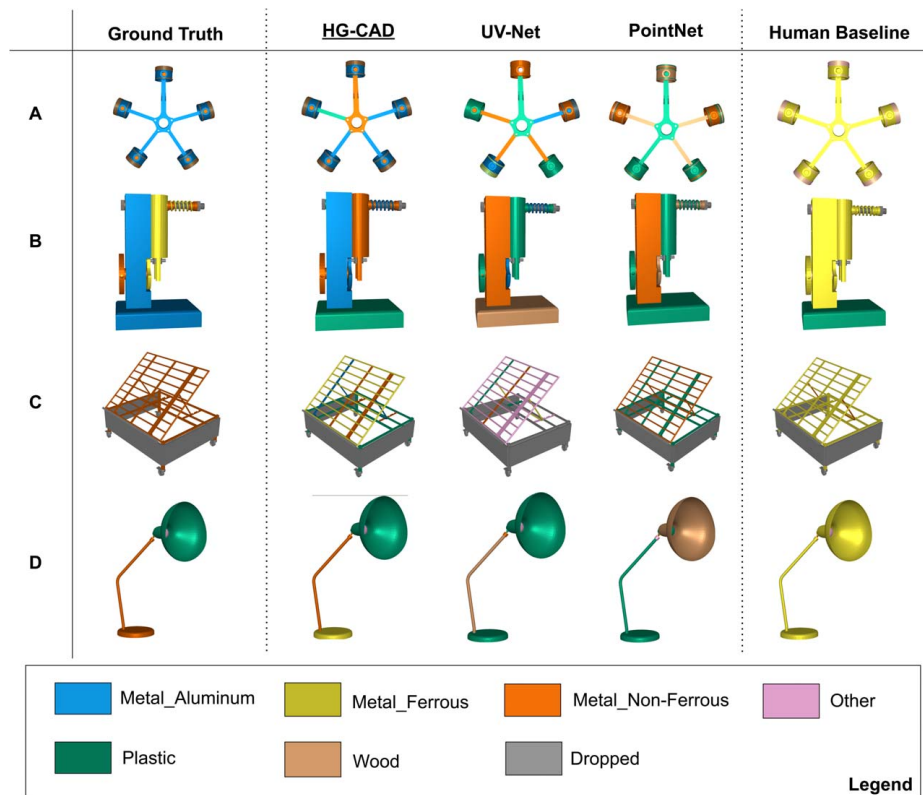


Fig. 7 Qualitative comparison of material prediction results for selected assemblies

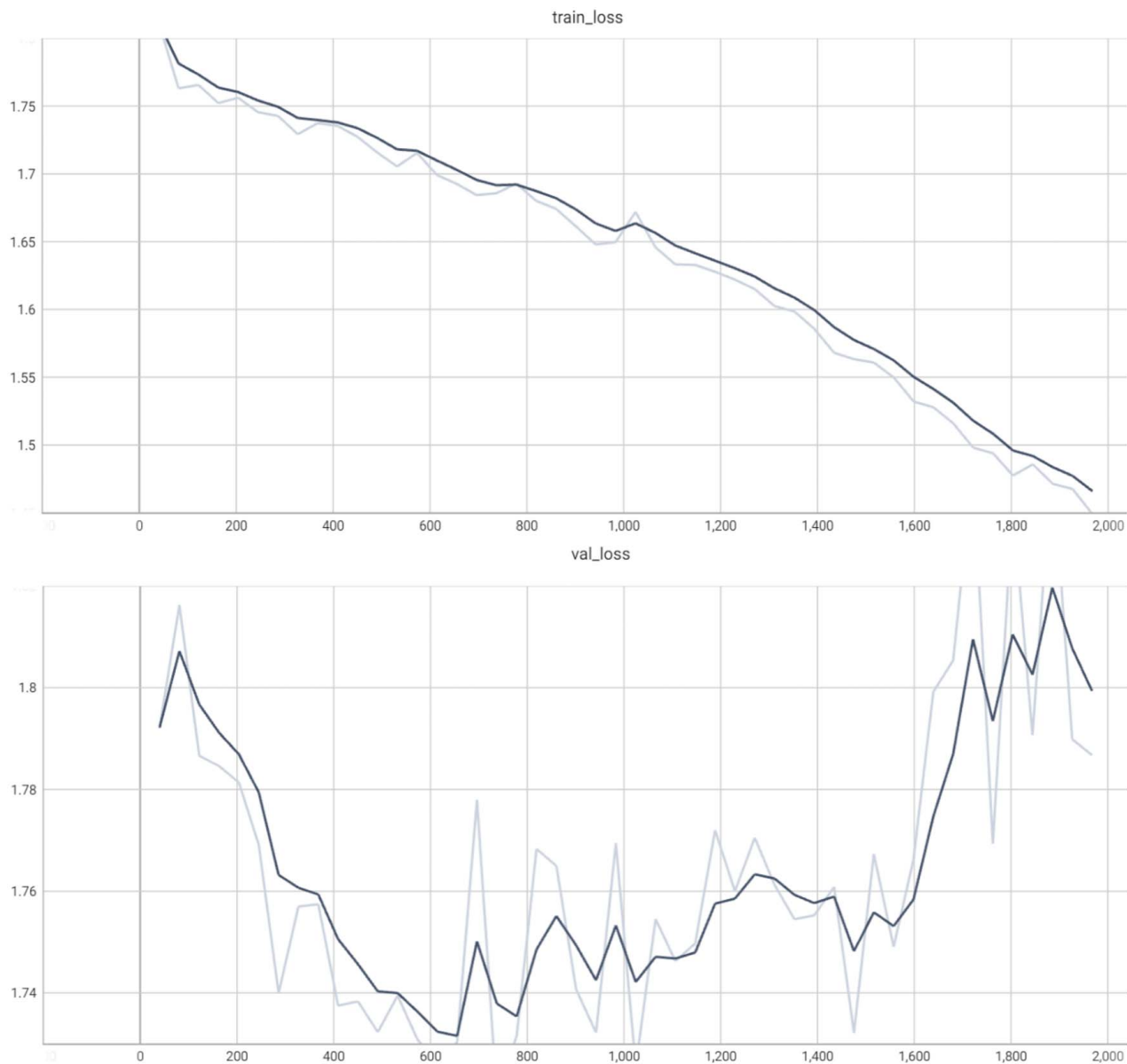
abduction, and induction reasoning. In addition, future work can be directed to experimenting with a more diverse set and combination of the neural network building layers. In the current iteration, we examined the performance with graph attention networks (GATs), graph isomorphism networks (GINs) as well as linear transformation layers as suggested in Ref. [31], and observed significant overfitting of the training dataset due to the uneven distribution of assembly architectures and material selections, as indicated in the convergence graph of Fig. 8. Furthermore, due to the highly variable discrepancies between the topology and structure of design assemblies, more experimental iterations should be performed with diverse splits. Similarly, we also intend to extend our human baseline sample size by incorporating human participants from different realms of engineering and expertise levels for a more comprehensive investigation and analysis of human engineers' prediction patterns.

**5.1.2 Functional and Behavioral Information.** The current stage of work focuses on the *structural* aspect of objects, including the geometric and physical properties of bodies and the assembly relationships that correlate them. While structural analysis has proven to be successful in the derivation and tabulation of material performance indices for standard mechanical design cases [69], the *functional* and *behavioral* aspects of the design are also highly influential to material selection [2,70]. Therefore, one promising improvement would be incorporating the functional information (i.e., the purpose) and the behavioral information (i.e., the attributes) of bodies as node features to provide more context for learning. Functional and behavioral information might be inferred from the name of parts in the assemblies, and prior work has shown promise in learning from these natural language labels commonly found in CAD [44,71], and large language models could be leveraged for transfer learning. While body-level semantic names might implicitly represent some functional and behavioral information, introducing additional user inputs (e.g., cost and size requirements) would enable predictions tailored to specific design requirements.

## 5.2 Future Plans

**5.2.1 Regression Model for Material Properties.** One limitation of the proposed work is its dependency on the Autodesk Fusion 360 Assembly Dataset and its categorical material library. Specifically, new designs from the Fusion 360 Gallery can be imported as additional data during training and inference. For a new assembly design acquired from the Autodesk Fusion 360 gallery, it can be encoded as a JSON file automatically by the FUSION 360 software along with the thumbnail images and 3D object files, where the embedding dimensions of the body-level features are automatically handled by the underlying UV-Net architecture. The dimensions of the graph embedding are determined by the hyperparameters defined by the user in the HG-CAD architecture. Despite alleviating possible confusion by dropping bodies of default and paint material, the regrouping process may introduce bias that limits the trained model's application to only materials found in the original dataset, thus limiting scalability and might not suit the needs of designers in different industries. A possible improvement might be to restructure the problem as a regression task and develop a model to predict relevant physical and mechanical properties involved in the selection of materials, expanding prior work that trained NNs directly on material properties [26,72]. Moreover, by mapping relevant material properties of the material library of the training data onto an Ashby chart, clustering the material properties on the chart may enable a more flexible material selection method.

**5.2.2 Graph Predictions and Similarity Search on Computer-Aided Design Models.** We envision the potential expansion of our work in producing diverse predictions for CAD automation that are not limited to material selection. Specifically, we seek to incorporate graph edge prediction and global context prediction functionalities by varying our current learning architecture. For *graph edge prediction task*, we aim to support the user-design process of organizing and correlating assembly bodies by providing insights into their hierarchical and relational information through



**Fig. 8 Convergence of HG-CAD**

representation learning based on their properties and that of the assembly. For *global context prediction*, we aim to provide users with an informative overview of the entire assembly design through representation learning tailored to the global graph features shared across individual bodies. Furthermore, the feature embedding of the entire assembly graph representation encoded with assembly-level topology and body-level geometrical information can be used with unsupervised learning methods, such as high-dimensional clustering, to enable *similarity search* on CAD models with structural and contextual awareness. One application scenario would be a post-design recommendation, in which we retrieve, based on similarity search results, past CAD designs that are similar to the newly produced user design, which may serve as sanity checks or baselines for the designer to reference. While the current solution only considered GraphSAGE, GAT [73], and GIN as the building layers of the graph encoder, we envision that additional GNN layer types provided by the DGL library, such as RelGraphConv and CFConv, can be incorporated in future work, provided the flexibility of the model architecture. Furthermore, we plan to seek further collaboration and a more diverse design repository access to apply our HG-CAD architecture. We believe

that our proposed architecture is flexible and competent enough to handle a wide range of use cases and anticipate that it may be applicable to high-value components and assemblies, satisfying the needs of broader users.

**5.2.3 A User-Guided Active Learning Process.** We want to overcome the limitations of learning solely from existing designs and would want to preserve the space for creativity by prompting a user-guided learning process in future works. Specifically, we envision an active learning process for the single-node prediction task, in which the model not only provides a single-shot prediction for the user-input design based on its learned knowledge from existing designs but also actively adjusts its weights and parameters as the user introduces new ground-truth information to existing bodies or even adding new bodies to the design. Another possible future direction to bring the user into the learning loop is to allow post-prediction feedback and adjustments. In the current iteration of the project, we evaluated the efficacy of the human labelers by collecting their feedback on the challenges during manual prediction, their confidence in their results, and how they think the entire experimental procedure can be improved. From the collective

feedback, we found that the major challenge that our human subjects encountered during their predictive labeling process derives from a lack of global context, which confounds their judgement on bodies of similar geometry. We hope this piece of information can demonstrate how the labeling of assembly part information can be hard to evaluate, thus highlighting our work's goal of automating the process by leveraging machine learning's capability of capturing diverse and latent representations from large repositories of past data to generalize better than human subjects. To provide more information that can bring the user into the loop, after displaying the top predicted material categories along with their corresponding confidence scores, we prompt the user with several questions, such as the requirement for the target assembly body's weight, volume, and cost, and adjust our predictions or re-ran the inference step based on the newly introduced information. This should effectively narrow the search space and thus predict a scope tailored to the user's specific needs.

## 6 Conclusion

In this paper, we propose HG-CAD, a unified architecture based on a hierarchical graph representation that enables intelligent computer-aided design by predicting the material category of bodies in design assemblies through joint learning of geometry and topology from both body-level and assembly-level scales in an end-to-end procedure, and with contextual awareness enabled through batch randomized node masking. Furthermore, we present a systematic workflow for the automated feature extraction, encoding, and graph representation generation of CAD models compatible with the predominant and scalable Fusion 360 Gallery Assembly Dataset.

We compare our proposed model with three state-of-the-art learning-based models and a human baseline for the experimental evaluation. Specifically, we formulate the body-level material prediction task as a node classification task on graphs, where we randomly generate the batch-level Boolean indicator masks to allow the network to make predictions on individual target nodes while referencing the ground-truth information from topologically neighboring nodes. Quantitative results demonstrated the effectiveness of the proposed methodology of representing CAD models as hierarchical graphs to enable joint learning of body-level geometry and assembly-level topology with contextual awareness in material prediction. Qualitative comparison using visualized predictions also reflected the potential of our proposed model in capturing and utilizing the similarity in geometry in conjunction with the topological symmetrical characteristics, showing promise in supporting human-in-the-loop design automation applications. Specifically, we believe that the proposed method's capacity to accommodate large-scale databases and flexibility in incorporating the designer's knowledge can be used to create a recommendation system for users by learning best practices from existing designs. Furthermore, the architecture may serve as a baseline or foundation for future works leveraging graph neural networks for design automation.

## Acknowledgment

This research was supported by the U.S. NASA MUREP Institutional Research Opportunity (MIRO) program (Award Number 80NSSC19M0200), donation from Autodesk Inc., and partially carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration (Award Number 80NM0018D0004).

## Conflict of Interest

There are no conflicts of interest.

## Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

## Nomenclature

AI	= artificial intelligence
ASME	= American Society of Mechanical Engineers
AIEDAM	= artificial intelligence for engineering design, analysis, and manufacturing
CC	= global clustering coefficient
CIE	= computers in engineering
CPL	= characteristic path length
CC-WS	= Watts-Strogatz local clustering coefficient
CPLRG	= characteristic path length of the equivalent random graph
DTM	= design theory and methodology
DP	= design parameter
FR	= functional requirement
GED	= graph edit distance
IFIP	= International Federation for Information Processing
ME	= mechanical engineering
SWI	= small-world index
TRL	= technology readiness levels

## References

- [1] Mouritz, A.P., 2012, *Introduction to Aerospace Materials*, Woodhead Publishing, Cambridge, UK.
- [2] Arnold, S. M., Cebon, D., and Ashby, M., 2012, *Materials Selection for Aerospace Systems*, National Aeronautics and Space Administration Glenn Research Center, Hanover, MD.
- [3] Miller, B.A., Shipley, R.J., Parrington, R.J., and Dennies, D.P., 2021, *Failure Analysis and Prevention*, Vol. 11, ASM International, Materials Park, OH, pp. 60–79.
- [4] Sargent, P., 1991, *Materials Information for CAD/CAM*, Butterworth-Heinemann, Oxford, UK.
- [5] Willis, K.D.D., Pu, Y., Luo, J., Chu, H., Du, T., Lambourne, J.G., Solar-Lezama, A., and Matusik, W., 2021, "Fusion 360 Gallery: A Dataset and Environment for Programmatic CAD Construction From Human Design Sequences," *ACM Trans. Graph.*, **40**(4), pp. 54–78.
- [6] Koch, S., Matveev, A., Jiang, Z., Williams, F., Artemov, A., Burnaev, E., Alexa, M., Zorin, D., and Panozzo, D., 2018, "ABC: A Big CAD Model Dataset for Geometric Deep Learning," The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, June 18–23.
- [7] Kim, S., Chi, H.-g., Hu, X., Huang, Q., and Ramani, K., 2020, "A Large-Scale Annotated Mechanical Components Benchmark for Classification and Retrieval Tasks With Deep Neural Networks," Proceedings of 16th European Conference on Computer Vision (ECCV), Glasgow, UK, Aug. 23–28.
- [8] Sarica, S., Luo, J., and Wood, K. L., 2020, "TechNet: Technology Semantic Network Based on Patent Data," *Expert Syst. Appl.*, **142**, p. 112995.
- [9] Willis, K. D. D., Jayaraman, P. K., Chu, H., Tian, Y., Li, Y., Grandi, D., Sanghi, A., et al., 2021, "JoinABLE: Learning Bottom-up Assembly of Parametric CAD Joints," 2021 Conference on Computer Vision and Pattern Recognition, Nashville, TN, June 19–25.
- [10] Jones, B., Hildreth, D., Chen, D., Baran, I., Kim, V., and Schulz, A., 2021, "Sb-gen: Structured BREP Graph Convolutional Network for Automatic Mating of CAD Assemblies," *CoRR*, pp. 1–16.
- [11] Lambourne, J.G., Willis, K.D., Jayaraman, P. K., Sanghi, A., Meltzer, P., and Shayani, H., 2021, "BRepNet: A Topological Message Passing System for Solid Models," Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, June 20–25, pp. 12773–2782.
- [12] Jayaraman, P. K., Sanghi, A., Lambourne, J., Davies, T., Shayani, H., and Morris, N., 2020, "Uv-net: Learning From Curve-Networks and Solids," IEEE / CVF Computer Vision and Pattern Recognition Conference (CVPR), Virtual, June 19–25.
- [13] Han, J., Forbes, H., Shi, F., Hao, J., and Schaefer, D., 2020, "A Data-Driven Approach for Creative Concept Generation and Evaluation," Proceedings of the Design Society: DESIGN Conference, Cavtat, Croatia, Oct. 26–29, Vol. 1, Cambridge University Press, pp. 167–176.
- [14] Sarica, S., Song, B., Luo, J., and Wood, K. L., 2021, "Idea Generation With Technology Semantic Network," *AI EDAM*, **35**(3), pp. 265–283.
- [15] Luo, J., Sarica, S., and Wood, K. L., 2021, "Guiding Data-Driven Design Ideation by Knowledge Distance," *Knowledge Based Syst.*, **218**, p. 106873.
- [16] Ashby, M. F., 1989, "Materials Selection in Conceptual Design," *Mater. Sci. Technol.*, **5**(6), pp. 517–525.



- [17] Ullah, N., Riaz, A. A., and Shah, S. A., 2020, "Investigation on Material Selection for the Columns of Universal Testing Machine (UTM) Using Granta's Design CES Eupack," *Tech. J.*, **25**(2), pp. 52–60.
- [18] Albiñana, J., and Vila, C., 2012, "A Framework for Concurrent Material and Process Selection During Conceptual Product Design Stages," *Mater. Des.*, **41**, pp. 433–446.
- [19] van Kesteren, I.E.H., Stappers, P.J., and de Bruijn, J.C.M., 2007, "Materials, User-Product Interaction and Other Decisions," *Int. J. Des.*, **1**(3), pp. 41–55.
- [20] Zarandi, M. H. F., Mansour, S., Hosseini, S. A., and Avazbeigi, M., 2011, "A Material Selection Methodology and Expert System for Sustainable Product Design," *Int. J. Adv. Manuf. Technol.*, **57**(9–12), pp. 885–903.
- [21] Bi, L., Zuo, Y., Tao, F., Liao, T. W., and Liu, Z., 2017, "Energy-Aware Material Selection for Product With Multicomponent Under Cloud Environment," *ASME J. Comput. Inf. Sci. Eng.*, **17**(3), p. 031007.
- [22] Abdullah, T. A., Popplewell, K., and Page, C. J., 2003, "A Review of the Support Tools for the Process of Assembly Method Selection and Assembly Planning," *Int. J. Prod. Res.*, **41**(11), pp. 2391–2410.
- [23] Odum, K., Soshi, M., and Yamazaki, K., 2022, "Numerical Study of Material Selection for Optimal Directed Energy Deposition Single Nozzle Powder Efficiency," *ASME J. Manuf. Sci. Eng.*, **144**(12), p. 121006.
- [24] Sirisalee, P., Parks, G.T., Ashby, M.F., and Clarkson, P.J., 2004, "Multi-Material Selection: Material Selection for Sandwich Beams," ASME 2004 International Design Engineering Technical Conferences and Information in Engineering Conference, Salt Lake City, UT, Sept. 28–Oct. 2, Vol. 46962, pp. 31–40.
- [25] Somkuwar, A., Khaira, H., and Somkuwar, V., 2010, "Materials Selection for Product Design Using Artificial Neural Network Technique," *J. Eng. Sci. Manag. Educ.*, **3**(2), pp. 51–54.
- [26] Chandrasekhar, A., Sridhara, S., and Suresh, K., 2022, "Integrating Material Selection with Design Optimization Via Neural Networks," *Eng. Comput.*, **38**, pp. 4715–4730.
- [27] Keiser, J.R., He, X., Sulejmanovic, D., Qu, J.N., Robb, K.R., and Oldinski, K., 2023, "Material Selection and Corrosion Studies of Candidate Bearing Materials for Use in Molten Chloride Salts," *ASME J. Sol. Energy Eng.*, **145**(2), p. 021001.
- [28] Eddy, D., Krishnamurthy, S., Grosse, I.R., and Wileden, J., 2014, "A Robust Surrogate Modeling Approach for Material Selection in Sustainable Design of Products," ASME 2014 International Design Engineering Technical Conferences and Information in Engineering Conference, Buffalo, NY, Aug. 17–20, p. V01AT02A091.
- [29] Dehghan-Manshadi, B., Mahmudi, H., Abedian, A., and Mahmudi, R., 2007, "A Novel Method for Materials Selection in Mechanical Design: Combination of Non-linear Normalization and a Modified Digital Logic Method," *Mater. Des.*, **28**(1), pp. 8–15.
- [30] Mamoon, A., Alhaji, A. U., and Abdullahi, I., 2021, "Application of Neural Network for Material Selection: A Review," *Int. J. Mater. Sci. Eng.*, **7**(2), pp. 1–16.
- [31] Zhou, C.-C., Yin, G.-F., and Hu, X.-B., 2009, "Multi-Objective Optimization of Material Selection for Sustainable Products: Artificial Neural Networks and Genetic Algorithm Approach," *Mater. Des.*, **30**(4), pp. 1209–1215.
- [32] Bian, S., Lin, T., Li, C., Fu, Y., Jiang, M., Wu, T., Hang, X., and Li, B., 2021, "Real-time Object Detection for Smart Connected Worker in 3D printing," 2021 International Conference on Computational Science (ICCS 2021), Krakow, Poland, June 16–18, M. Paszynski, D. Kranzlmüller, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Slood, eds., Springer International Publishing, pp. 554–567.
- [33] Girshick, R., Donahue, J., Darrell, T., and Malik, J., 2014, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, June 23–28.
- [34] Li, Y., Tarlow, D., Brockschmidt, M., and Zemel, R., 2016, "Gated Graph Sequence Neural Networks," 4th International Conference on Learning Representations (ICLR 2016), San Juan, Puerto Rico, May 2–4.
- [35] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E., 2017, "Neural Message Passing for Quantum Chemistry," International Conference on Machine Learning, Sydney, Australia, Aug. 6–11, pp. 1263–1272.
- [36] Kipf, T. N., and Welling, M., 2017, "Semi-Supervised Classification With Graph Convolutional Networks," 5th International Conference on Learning Representations (ICLR 2017), Toulon, France, Apr. 24–26.
- [37] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y., 2018, "Graph Attention Networks," 6th International Conference on Learning Representations (ICLR 2018), Vancouver, BC, Canada, Apr. 30–May 3.
- [38] Xu, K., Hu, W., Leskovec, J., and Jegelka, S., 2019, "How Powerful Are Graph Neural Networks?" 7th International Conference on Learning Representations (ICLR 2019), New Orleans, LA, May 6–9.
- [39] Khasahmadi, A. H., Hassani, K., Moradi, P., Lee, L., and Morris, Q., 2020, "Memory-Based Graph Networks," 8th International Conference on Learning Representations (ICLR 2020), Addis Ababa, Ethiopia, Apr. 26–30.
- [40] Xing, Y., He, T., Xiao, T., Wang, Y., Xiong, Y., Xia, W., Wipf, D., Zhang, Z., and Soatto, S., 2021, "Learning Hierarchical Graph Neural Networks for Image Clustering," 2021 International Conference on Computer Vision (ICCV 2021), Montreal, Canada, Oct. 11–17.
- [41] Tao, F., Sui, F., Liu, A., Qi, Q., Zhang, M., Song, B., Guo, Z., Lu, S. C.-Y., and Nie, A. Y. C., 2019, "Digital Twin-Driven Product Design Framework," *Int. J. Prod. Res.*, **57**(12), pp. 3935–3953.
- [42] Feng, Y., Zhao, Y., Zheng, H., Li, Z., and Tan, J., 2020, "Data-Driven Product Design Toward Intelligent Manufacturing: A Review," *Int. J. Adv. Rob. Syst.*, **17**(2), p. 1729881420911257.
- [43] Funkhouser, T., Kazhdan, M., Shilane, P., Min, P., Kiefer, W., Tal, A., Rusinkiewicz, S., and Dobkin, D., 2004, "Modeling by Example," *ACM Trans. Graph.*, **23**(3), pp. 652–663.
- [44] Ferrero, V., DuPont, B., Hassani, K., and Grandi, D., 2022, "Classifying Component Function in Product Assemblies With Graph Neural Networks," *ASME J. Mech. Des.*, **144**(2), p. 021406.
- [45] Williams, G., Meisel, N. A., Simpson, T. W., and McComb, C., 2019, "Design Repository Effectiveness for 3D Convolutional Neural Networks: Application to Additive Manufacturing," *ASME J. Mech. Des.*, **141**(11), p. 111701.
- [46] Wang, Y., Grandi, D., Cui, D., Rao, V., and Goucher-Lambert, K., 2021, "Understanding Professional Designers' Knowledge Organization Behavior: A Case Study in Product Teardowns," ASME 2021 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Virtual, Aug. 17–19, Vol. 85420, American Society of Mechanical Engineers, p. V006T06A046.
- [47] Chaudhuri, S., Ritchie, D., Wu, J., Xu, K., and Zhang, H., 2020, "Learning Generative Models of 3D Structures," *Comput. Graph. Forum*, **39**(2), pp. 643–666.
- [48] Li, J., Niu, C., and Xu, K., 2020, "Learning Part Generation and Assembly for Structure-Aware Shape Synthesis," 34th AAAI Conference on Artificial Intelligence (AAAI 2020), New York, NY, Feb. 7–12.
- [49] Mo, K., Guerrero, P., Yi, L., Su, H., Wonka, P., Mitra, N.J., and Guibas, L.J., 2019, "StructureNet: Hierarchical Graph Networks for 3d Shape Generation," *ACM Transactions on Graphics*, **38**(6), pp. 1–19.
- [50] Slyadnev, S., Malyshev, A., Voevodin, A., and Turlapov, V., 2020, "On the Role of Graph Theory Apparatus in a CAD Modeling Kernel," 30th International Conference on Computer Graphics and Machine Vision, Saint Petersburg, Russia, Sept. 22–25.
- [51] Huang, H., Lo, S., Zhi, G., and Yuen, R., 2008, "Graph Theory-Based Approach for Automatic Recognition of CAD Data," *Eng. Appl. Artif. Intell.*, **21**(7), pp. 1073–1079.
- [52] Sunil, V., Agarwal, R., and Pande, S., 2010, "An Approach to Recognize Interacting Features From B-Rep CAD Models of Prismatic Machined Parts Using a Hybrid (Graph and Rule Based) Technique," *Comput. Ind.*, **61**(7), pp. 686–701.
- [53] Jones, B., Hildreth, D., Chen, D., Baran, I., Kim, V. G., and Schulz, A., 2021, "Automate: A Dataset and Learning Approach for Automatic Mating of CAD Assemblies," *ACM Trans. Graph.*, **40**(6), pp. 1–18.
- [54] Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., and Battaglia, P. W., 2021, "Learning Mesh-Based Simulation With Graph Networks," 9th International Conference on Learning Representations (ICLR 2021), Virtual, May 3–7.
- [55] Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., and Zhou, J., 2019, "Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks," *arXiv: Learning*, pp. 1–18.
- [56] Fey, M., and Lenssen, J. E., 2019, "Fast Graph Representation Learning with Geometric," 7th International Conference on Learning Representations (ICLR 2019), New Orleans, LA, May 6–9.
- [57] Hamilton, W., Ying, Z., and Leskovec, J., 2017, "Inductive Representation Learning on Large Graphs," 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, Dec. 4–9, pp. 1024–1034.
- [58] Maas, A. L., Hannun, A. Y., and Ng, A. Y., 2013, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," 30th International Conference on Machine Learning (ICML 2013), Atlanta, GA, June 16–21.
- [59] Xu, K., Li, C., Tian, Y., Sonobe, T., Kawarabayashi, K.-i., and Jegelka, S., 2018, "Representation Learning on Graphs With Jumping Knowledge Networks," 35th International Conference on Machine Learning, Stockholm, Sweden, July 10–15, PMLR, pp. 5453–5462.
- [60] Ioffe, S., and Szegedy, C., 2015, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," 32nd International Conference on Machine Learning, Lille, France, July 6–11, PMLR, pp. 448–456.
- [61] He, K., Zhang, X., Ren, S., and Sun, J., 2015, "Delving Deep Into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," 2015 International Conference on Computer Vision, Santiago, Chile, Dec. 11–15.
- [62] Qi, C. R., Su, H., Mo, K., and Guibas, L. J., 2016, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," 2016 Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, June 26–July 1.
- [63] Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. G., 2015, "Multi-View Convolutional Neural Networks for 3d Shape Recognition," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, Dec. 13–16.
- [64] Kingma, D. P., and Ba, J., 2017, "Adam: A Method for Stochastic Optimization," 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, May 7–9.
- [65] Loshchilov, I., and Hutter, F., 2017, "SGDR: Stochastic Gradient Descent with Warm Restarts," 5th International Conference on Learning Representations (ICLR 2017), Toulon, France, Apr. 24–26.
- [66] He, K., Zhang, X., Ren, S., and Sun, J., 2016, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, June 27–30.
- [67] Zhao, T., Liu, Y., Neves, L., Woodford, O., Jiang, M., and Shah, N., 2021, "Data Augmentation for Graph Neural Networks," Proceedings of the AAAI Conference on Artificial Intelligence, Hong Kong, China, Feb. 2–9, Vol. 35, pp. 11015–11023.
- [68] Wang, P., 2007, "From Nars to a Thinking Machine," Proceedings of the 2007 Conference on Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms: Proceedings of the AGI Workshop 2006, Amsterdam, Netherlands, June 7.

- [69] Ashby, M. F., 2011, *Materials Selection in Mechanical Design*, 4th ed., Butterworth-Heinemann, Burlington, MA.
- [70] Cheng, Z., and Ma, Y., 2017, "Explicit Function-Based Design Modelling Methodology With Features," *J. Eng. Des.*, **28**(1), pp. 1–27.
- [71] Bian, S., Grandi, D., Hassani, K., Sadler, E., Borijin, B., Fernandes, A., Wang, A., Lu, T., Otis, R., Ho, N., and Li, B., 2022, "Material Prediction for Design Automation Using Graph Representation Learning," ASME 2022 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE 2022), St. Louis, MO, Aug. 14–17.
- [72] Merayo, D., Rodríguez-Prieto, A., and Camacho, A.M., 2020, "Prediction of Physical and Mechanical Properties for Metallic Materials Selection Using Big Data and Artificial Neural Networks," *IEEE Access*, **8**, pp. 13444–13456.
- [73] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y., 2018, "Graph Attention Networks," 6th International Conference on Learning Representations (ICLR 2018), Vancouver, Canada, Apr. 30–May 3.