CONTRASTIVE LEARNING TO ALIGN DRAWINGS AND CAD MODELS FOR RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Retrieving computer-aided design (CAD) models from drawings can support the discovery and reuse of existing designs in engineering workflows. However, differences between visual depictions and parametric model representations make it challenging to identify the relevant CAD model, particularly when the query image has been altered. This paper proposes a contrastive learning framework that aligns CAD drawings with CAD model representations in a shared embedding space. The framework learns correspondences between drawings and model construction sequences, enabling retrieval through embedding similarity. Beyond matching an original drawing to its corresponding CAD model, the approach also retrieves relevant models from altered query images. We evaluate the framework on a dataset of parametric CAD command sequences derived from the ABC dataset, assessing retrieval performance under both original and altered query conditions. Experimental results show improved retrieval performance compared with baseline methods, demonstrating that contrastive learning can effectively align CAD drawings with parametric CAD model representations. These findings support the applicability of contrastive learning beyond image–text alignment to image–CAD alignment, enabling CAD model retrieval directly from visual queries.

1 INTRODUCTION

Computer-aided design (CAD) models are central to modern engineering workflows and provide structured representations of both object geometry and the operations used to construct it. As CAD repositories continue to grow, effective retrieval becomes increasingly important for design reuse, automated modelling, and engineering knowledge discovery. In many practical settings, however, a designer may have only a drawing, image, or rendered representation of a desired component rather than the corresponding CAD model. This motivates image-to-CAD retrieval, where a visual query is used to identify a corresponding CAD model from a repository.

This problem is challenging because the query and target belong to different representation domains. An image captures the visual appearance of an object, whereas a parametric CAD model can be represented as a structured sequence of modelling operations and parameters (Wu et al., 2021; Alam & Ahmed, 2025). A retrieval system must therefore bridge visual information and sequential CAD construction information while remaining sufficiently discriminative to distinguish between individual models.

Contrastive representation learning provides a natural framework for this cross-modal alignment problem. Contrastive Language-Image Pre-training (CLIP) (Radford et al., 2021), for example, aligns heterogeneous image and text representations within a shared embedding space by increasing the similarity of corresponding samples while separating non-corresponding ones. We extend this principle to image-to-CAD retrieval by processing visual queries and parametric CAD command sequences with modality-specific encoders and aligning their representations in a common embedding space.

A further challenge arises when the query does not exactly reproduce the original CAD model. In practical design-reuse scenarios, a user may possess a modified version of an existing component and seek the source CAD model as a reusable starting point. GenCAD (Alam & Ahmed, 2025) provides canonical renderings together with geometrically altered variants associated with the same

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

source designs. This allows us to investigate whether a learned image-to-CAD representation can retain the association between an altered visual query and its source parametric CAD model.

In this work, we develop a dual-encoder contrastive framework for retrieving parametric CAD models directly from visual queries. Query images are encoded using a pretrained CLIP RN50 visual encoder, while CAD models are represented by their parametric construction sequences and encoded using a Transformer-based sequence encoder. The resulting embeddings are aligned using a symmetric contrastive objective. At inference time, a query image is embedded and compared with precomputed CAD gallery embeddings using cosine similarity to rank candidate models.

We evaluate the framework using parametric CAD command sequences and paired visual queries following the GenCAD data organization (Alam & Ahmed, 2025), derived from the ABC dataset (Koch et al., 2019). The evaluation considers both canonical renderings and geometrically altered variants, enabling us to study direct image-to-CAD retrieval as well as recovery of the source CAD representation from a modified design.

The main contributions of this work are threefold:

- **A contrastive image-to-CAD retrieval framework** that aligns visual queries and CAD command sequences within a shared embedding space using modality-specific encoders.
- **An evaluation setting with canonical and altered queries** that measures both direct retrieval and recovery of the source CAD model from geometrically modified variants.
- **An empirical evaluation of cross-modal CAD retrieval** demonstrating strong performance and examining the robustness of the learned representation to query modification.

Together, these contributions investigate whether visual representations can be aligned directly with parametric CAD construction sequences for retrieval and whether this correspondence can be preserved when the visual query represents a modified version of the source design.

2 Related Work

2.1 CAD and 3D Model Retrieval

Prior work on 3D and CAD retrieval has explored several representations, including rendered views, meshes, point clouds, voxels, and boundary-representation graphs. Multi-view methods such as MVCNN (Su et al., 2015) represent a 3D object using multiple rendered images and aggregate image features into a compact shape descriptor. Such approaches demonstrate that two-dimensional visual features can provide effective representations for 3D shape retrieval, but the database models are ultimately represented through their visual appearance rather than their parametric construction history.

Graph-based CAD retrieval instead operates directly on structured geometric representations. Methods based on STEP or B-Rep models construct graphs whose nodes and edges encode geometric entities and their connectivity, allowing graph neural networks to learn descriptors that preserve geometric and topological information. Recent approaches such as GC-CAD (Quan et al., 2024) and CADGCL (Qin et al., 2025) further apply self-supervised or contrastive graph learning to mechanical CAD retrieval. These methods exploit structural information available in CAD models, but generally assume that the query itself is represented as a CAD model or graph.

Sketch-based retrieval provides a closer setting to our work because the query is two-dimensional. Existing methods have aligned sketches with projected CAD views using structured representations such as loop-relation trees or learned latent descriptors (Qin et al., 2022). However, the retrieved CAD models are typically represented through projected geometry or view-derived structures. Our work instead retrieves CAD models represented directly by their parametric construction sequences, creating a cross-modal correspondence between visual queries and structured modelling operations.

2.2 Parametric CAD Representations

Parametric CAD models can be represented as ordered sequences of modelling commands and associated parameters, providing information about how a model is constructed in addition to its final

geometry. DeepCAD (Wu et al., 2021) introduced a sequence representation based on sketch primitives and extrusion operations and used a Transformer autoencoder to learn latent representations of CAD construction sequences. This representation has subsequently been adopted in later parametric CAD generation frameworks (Alam & Ahmed, 2025).

Our work uses this command-sequence representation as the retrieval-side representation of a CAD model. Rather than converting gallery models into rendered views, meshes, or graphs, the CAD encoder operates directly on their modelling command sequences. This allows visual queries to be compared with structured representations that retain information about modelling operations and their parameters.

2.3 Contrastive Cross-Modal Learning

Contrastive learning constructs representation spaces in which corresponding samples are encouraged to have similar embeddings while non-corresponding samples are separated. SimCLR (Chen et al., 2020) demonstrated the effectiveness of this principle for self-supervised visual representation learning, while CLIP (Radford et al., 2021) extended it to cross-modal image–text alignment using separate image and text encoders trained with a symmetric contrastive objective.

The same principle can be applied when the second modality is a structured representation rather than natural language. In the CAD domain, GenCAD (Alam & Ahmed, 2025) uses contrastive learning to align rendered image features with representations derived from parametric CAD command sequences as part of an image-conditioned CAD generation pipeline. Our work builds on this cross-modal alignment principle but focuses specifically on retrieval: a visual query is compared directly against a gallery of parametric CAD sequence embeddings. We evaluate this formulation as large-gallery image-to-CAD retrieval and additionally examine retrieval when the query is a geometrically altered variant associated with the source CAD model.

3 Methodology

3.1 Problem Formulation

We formulate image-to-CAD retrieval as a cross-modal ranking problem. Given a query image x and a gallery of parametric CAD models

$$\mathcal{G} = \{C_1, \ldots, C_N\}, \tag{1}$$

the objective is to rank the gallery such that the CAD model associated with the query appears as highly as possible.

The proposed framework follows a dual-encoder architecture with an image encoder f_{img} and a CAD sequence encoder f_{cad}. The two encoders map their respective inputs into a common $d = 512$ dimensional embedding space. Their outputs are L_2-normalized,

$$\mathbf{u}(x) = \frac{f_{\text{img}}(x)}{\|f_{\text{img}}(x)\|_2}, \qquad \mathbf{v}(C) = \frac{f_{\text{cad}}(C)}{\|f_{\text{cad}}(C)\|_2}. \tag{2}$$

The encoders are trained jointly so that corresponding image–CAD pairs have high similarity while non-corresponding pairs are separated.

3.2 Parametric CAD Representation

Each CAD model is represented using the parametric command-sequence representation adopted from DeepCAD and GenCAD (Wu et al., 2021; Alam & Ahmed, 2025). A CAD model is expressed as an ordered sequence

$$C = (\mathbf{q}_1, \ldots, \mathbf{q}_L), \qquad \mathbf{q}_i = [c_i, a_{i,1}, \ldots, a_{i,16}], \tag{3}$$

where c_i denotes a command type and $a_{i,r}$ denotes the value of argument slot r. Each sequence element corresponds to a single modelling step. The command token specifies the type of operation, while the associated argument slots store the parameters required by that operation. For example, sketch commands encode geometric primitives and their parameters, whereas an extrusion command

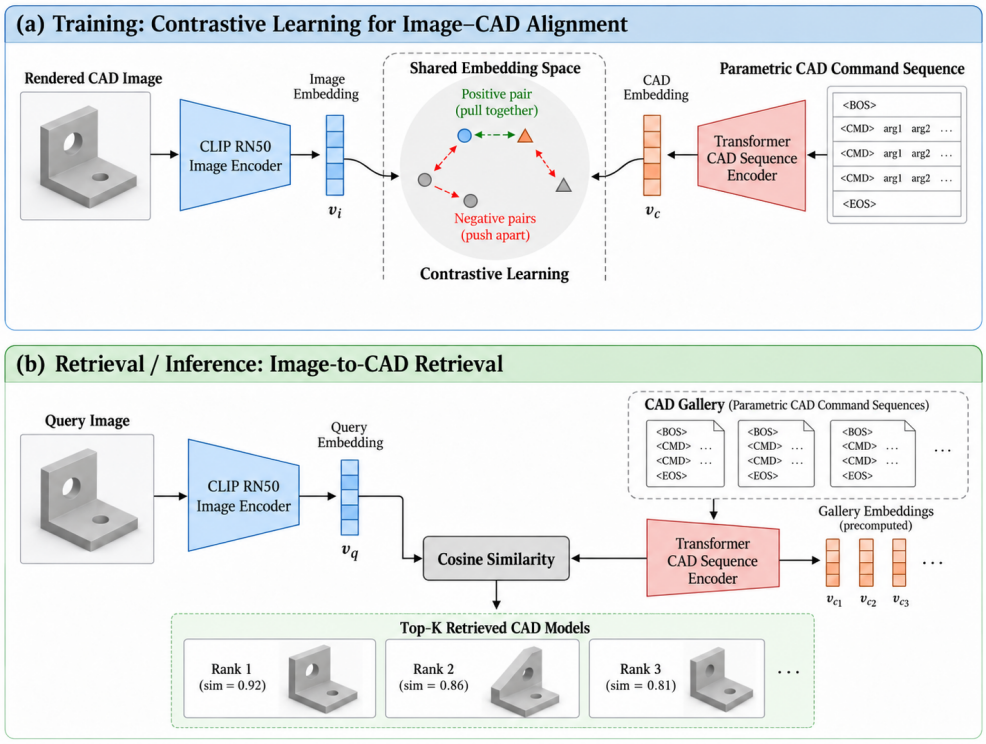

Figure 1: Overview of the image-to-CAD retrieval framework. During training, rendered CAD images and their corresponding parametric CAD command sequences are encoded using a pretrained CLIP RN50 image encoder and a Transformer-based CAD sequence encoder. A symmetric contrastive objective aligns both modalities in a shared embedding space. At inference time, the query image embedding is compared with precomputed CAD gallery embeddings using cosine similarity.

describes how the resulting sketch geometry is extended into three dimensions. The complete sequence therefore provides a structured description of how the CAD model is constructed rather than representing only its final shape.

The command vocabulary contains six operations: `Line`, `Arc`, `Circle`, `SOL`, `Ext`, and `EOS`. Continuous parameters are quantized into 256 discrete levels, while unused argument positions are represented by a dedicated padding value. Sequences are padded or truncated to a maximum length of 64 commands, and positions following the first `EOS` command are masked during sequence encoding.

3.3 Dual-Encoder Architecture

Figure 1 summarizes the proposed training and retrieval framework.

Image encoder. The visual branch uses the pretrained CLIP RN50 image encoder (Radford et al., 2021). Its output is mapped to the 512-dimensional shared embedding space and L_2-normalized. The visual encoder is fine-tuned jointly with the CAD branch during training.

CAD sequence encoder. The CAD branch is a Transformer encoder (Vaswani et al., 2017) trained from scratch. For each row, the command identifier and its quantized argument values are embedded and combined to form a command representation. A learned classification token is prepended to the sequence, and learned positional embeddings are added before Transformer encoding.

The CAD encoder contains four Transformer layers, eight attention heads, and a hidden dimension of 512. The output corresponding to the classification token is used as the global CAD representation and is L_2-normalized before cross-modal comparison.

3.4 Contrastive Training and Retrieval

Given a mini-batch of B paired image–CAD samples $\{(x_i, C_i)\}_{i=1}^{B}$, let $\mathbf{u}_i$ and $\mathbf{v}_j$ denote their normalized image and CAD embeddings. Pairwise logits are computed as

$$\ell_{ij} = \alpha\, \mathbf{u}_i^\top \mathbf{v}_j, \tag{4}$$

where α is a learned logit scale initialized to $1/0.07$.

Training follows the symmetric CLIP objective (Radford et al., 2021). Cross-entropy is applied in both image-to-CAD and CAD-to-image directions,

$$\mathcal{L} = \frac{1}{2}\left[\mathrm{CE}(\mathbf{L}, \mathbf{y}) + \mathrm{CE}(\mathbf{L}^\top, \mathbf{y})\right], \tag{5}$$

where $\mathbf{L} = [\ell_{ij}]$ and $\mathbf{y} = (1, \ldots, B)$ represents the diagonal image–CAD correspondences. The reported model is trained using this objective without an additional CAD-specific similarity loss.

At inference time, the embedding of every CAD model in the retrieval gallery is computed once and stored. A query image is encoded independently and compared with all gallery embeddings. Since both modalities are L_2-normalized, their dot product is equivalent to cosine similarity. Gallery models are ranked in descending similarity order, and retrieval performance is evaluated using the sequence-equivalent Recall@K criterion defined in Section 4.2.

4 Experimental Setup

4.1 Dataset and Training

We use parametric CAD command sequences and rendered images following the GenCAD data organization (Alam & Ahmed, 2025) derived from the ABC dataset (Koch et al., 2019). Only samples containing both a valid CAD command sequence and at least one rendered image are retained. After filtering, the dataset contains 152,530 training models, 8,515 validation models, and 7,629 held-out test models. The official training, validation, and test partitions remain disjoint.

Each source CAD model is associated with a rendering of the original model and additional renderings of slightly modified variants (Alam & Ahmed, 2025). We refer to the rendering of the original, unmodified CAD model as view 0. The remaining GenCAD image indices correspond to geometrically altered variants derived from the same source model; therefore, "view" refers to the GenCAD image index and not simply to a different camera viewpoint. During training, one available rendered image is sampled for each source CAD model and paired with its parametric command sequence.

The image encoder is initialized from the pretrained CLIP RN50 visual backbone (Radford et al., 2021), while the CAD Transformer is trained from scratch. Images are processed at 224×224 resolution. The model is trained for 150 epochs using AdamW (Loshchilov & Hutter, 2019) with a learning rate of 10^{-4}, batch size 512, and random seed 42. Both encoders are optimized jointly using the symmetric contrastive objective described in Section 3.4. Model selection is based on validation loss; the selected checkpoint for the reported run corresponds to epoch 149.

4.2 Evaluation Protocol

Evaluation is performed as image-to-CAD retrieval over a fixed gallery containing all 7,629 CAD models in the held-out test partition. We conduct two experiments with distinct retrieval objectives.

Experiment 1: Canonical image-to-CAD retrieval. The first experiment evaluates direct recovery of a CAD model from its canonical rendering. For each test instance, view 0, corresponding to the original unmodified CAD model, is used as the query image. A total of 7,609 test models contain a valid view-0 rendering. The objective is to retrieve the parametric CAD representation corresponding to the object depicted in the query image from the complete test gallery.

Experiment 2: Retrieval from an altered design. The second experiment evaluates whether the source CAD model can still be recovered when the visual query represents a modified version of that design. For each of the 7,629 test models, one available image with a GenCAD index between

Table 1: Image-to-CAD retrieval performance on the held-out test set. Recall@K treats CAD instances with identical quantized parametric command sequences as equivalent retrieval targets.

Query condition	# Queries	Gallery	R@1 (%)	R@5 (%)	R@10 (%)
Canonical (view 0)	7,609	7,629	69.60	89.93	93.86
Canonical/altered (view 0–4)	7,629	7,629	67.40	89.54	93.72

0 and 4 is selected using a fixed random seed. This produces a query set containing both canonical renderings and geometrically altered variants. When an altered variant is selected, the retrieval target remains the parametric representation of the source CAD model from which that variant was derived. This experiment therefore evaluates whether the learned embedding can associate a modified design with the original CAD model that may serve as a reusable starting point.

Candidates are ranked using cosine similarity between the normalized query embedding and precomputed CAD gallery embeddings. Performance is reported as Recall@K for $K \in \{1, 5, 10\}$. Different dataset identifiers can correspond to identical quantized CAD command sequences; therefore, a retrieval is considered correct if any of the top-K candidates has a command sequence identical to that of the target CAD model up to and including the first `EOS` command.

5 Results

5.1 Overall Retrieval Performance

We evaluate the proposed image-to-CAD retrieval framework on the held-out test partition using the sequence-equivalent Recall@K criterion described in Section 4.2. The checkpoint with the lowest validation loss is used for evaluation; for the reported run, this corresponds to epoch 149. In both experiments, retrieval is performed against the fixed gallery containing all 7,629 test CAD models.

Table 1 reports the results for canonical and canonical/altered queries.

Canonical image-to-CAD retrieval. The first experiment evaluates direct recovery of the parametric CAD model from its canonical rendering. Of the 7,629 gallery models, 7,609 contain a valid view-0 image and are used as queries. The proposed framework achieves 69.60% R@1, 89.93% R@5, and 93.86% R@10. Thus, a sequence-equivalent CAD representation is returned as the highest-ranked result for approximately seven out of ten queries and within the top ten for more than 93% of canonical queries. The increase from R@1 to R@5 is particularly relevant for CAD search: the target is recovered within only five candidates for almost 90% of queries, reducing a gallery of thousands of CAD models to a small set of highly relevant candidates.

Source CAD retrieval from altered queries. The second experiment evaluates whether the source CAD representation remains retrievable when the query may depict a geometrically altered version of the design. In GenCAD, view 0 represents the original CAD model, while additional image indices correspond to altered variants derived from the same source model (Alam & Ahmed, 2025). For each of the 7,629 test models, one available image with an index between 0 and 4 is selected using a fixed random seed, producing a mixture of canonical and altered queries. Regardless of the selected image, the retrieval target remains the parametric representation of the corresponding source CAD model.

Under this condition, the framework achieves 67.40% R@1, 89.54% R@5, and 93.72% R@10. Relative to canonical retrieval, performance decreases by only 2.20 percentage points at R@1, 0.39 points at R@5, and 0.14 points at R@10. The reduction is therefore concentrated primarily at rank 1, while retrieval at larger depths remains almost unchanged. This suggests that geometric modification can alter the ordering of the nearest CAD candidates without substantially removing the source representation from the high-similarity neighborhood. From a design-reuse perspective, a modified design can therefore still lead the retrieval system back to the source parametric CAD model from which it was derived.

Table 2: Contextual comparison with representative CAD retrieval results. Evaluation protocols differ across methods and the numbers are therefore not directly comparable.

Method / query	**Evaluation scale**	**R@1**	**R@5**	**R@10**
GenCAD, image (Alam & Ahmed, 2025)	2,048 sampled	60.77	–	–
CAD-RAG, sketch (Ananthakrishnan et al., 2025)	403 queries	62.78	85.61	93.05
OrthoCAD, 3-view (Mahajan et al., 2025)	∼42.5k test models	94.51	96.71	–
MM-CAD, image (Bharathi et al., 2026)	19,263 gallery	33.20	55.24	64.20
Ours, canonical	7,609 / 7,629 full gallery	69.60	89.93	93.86
Ours, canonical/altered	7,629 / 7,629 full gallery	67.40	89.54	93.72

Overall, the model recovers a sequence-equivalent CAD representation within the top ten for more than 93% of queries in both conditions, while retaining R@1 above 67% even when the query set includes geometrically altered designs.

5.2 Comparison with Prior CAD Retrieval Work

Direct numerical comparison between CAD retrieval methods is difficult because published approaches differ in query modality, CAD representation, evaluation scale, and relevance definition. We therefore report representative published results in Table 2 as contextual comparisons rather than controlled baselines.

GenCAD (Alam & Ahmed, 2025) is the closest prior work to our representation-learning setting because it also aligns rendered images with parametric CAD programs using contrastive learning. Its retrieval evaluation, however, uses sampled candidate pools rather than ranking against a fixed full gallery. At its largest reported pool of 2,048 candidates, GenCAD achieves 60.77% top-1 accuracy, whereas our canonical experiment ranks each query against all 7,629 held-out CAD models and achieves 69.60% R@1.

CAD-RAG (Ananthakrishnan et al., 2025) evaluates retrieval from genuine hand-drawn sketches, while MM-CAD (Bharathi et al., 2026) evaluates photorealistic image queries against a 19,263-model B-Rep gallery. OrthoCAD (Mahajan et al., 2025) instead uses three structured orthographic views and reports substantially higher retrieval accuracy. These methods therefore address different query domains and CAD representations, preventing direct performance conclusions from their numerical results.

Other CAD retrieval work also uses substantially different evaluation criteria. CADSketchNet (Manda et al., 2021) evaluates category-level retrieval, CADGCL (Qin et al., 2025) reports CAD-to-CAD retrieval using mAP, GenCAD-3D (Yu et al., 2026) uses point clouds or meshes as queries, and recent text-to-CAD work (Pan et al., 2026) evaluates retrieval from natural language.

Overall, the comparison highlights the importance of interpreting retrieval accuracy together with query modality, gallery construction, and correctness criteria. Under our fixed 7,629-model gallery, the proposed framework achieves 69.60% R@1 and 93.86% R@10 for canonical queries, while the mixed canonical/altered condition retains 67.40% R@1 and 93.72% R@10. A controlled comparison would require evaluating competing retrieval methods under the same test split, gallery, query images, and sequence-equivalent correctness criterion.

5.3 Qualitative Retrieval Analysis

Figure 2 provides representative examples of the ranked retrieval results. The examples include a successful rank-1 retrieval, a case in which the exact ground-truth instance appears at rank 2, and more challenging cases in which the exact ground-truth instance is ranked outside the top five.

The qualitative visualization reports the rank of the exact ground-truth dataset instance, whereas the quantitative Recall@K metric treats CAD instances with identical parametric command sequences as equivalent. The displayed ranks are therefore a stricter instance-level visualization and should primarily be interpreted as examples of how the learned embedding organizes CAD geometry.

In Figure 2(b), the query is directly matched to its corresponding CAD model at rank 1. Figure 2(d) illustrates a near-miss case in which the exact ground-truth instance appears at rank 2. Although the

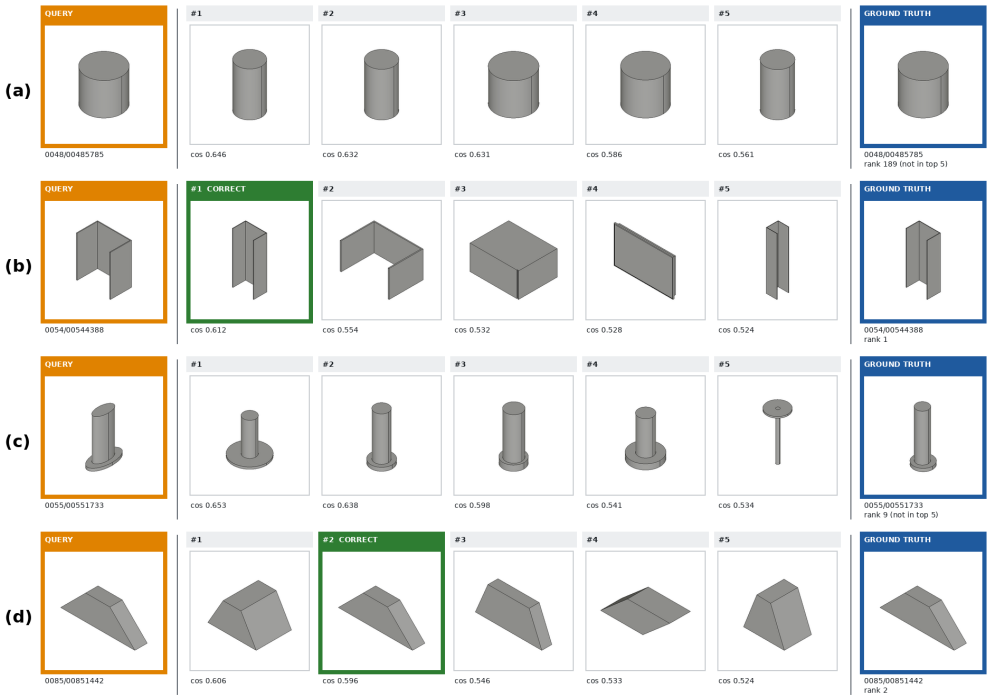

Figure 2: Qualitative image-to-CAD retrieval examples. Each row shows a query image, the top five retrieved CAD models, and the exact ground-truth CAD instance. (a) Challenging case with the ground-truth instance at rank 189. (b) Successful rank-1 retrieval. (c) Challenging case with the ground-truth instance at rank 9. (d) Near-miss case with the ground-truth instance at rank 2.

highest-ranked candidate differs from the ground truth, it preserves a closely related global shape, indicating that geometrically similar models occupy nearby regions of the learned embedding space.

The more difficult examples exhibit the same general behavior. In Figure 2(a), the exact ground-truth instance is ranked 189th, while the highest-ranked candidates remain cylindrical and preserve the broad primitive structure of the query. Their primary differences concern dimensions and proportions such as height and diameter. Similarly, in Figure 2(c), the exact target appears at rank 9, while the highest-ranked candidates preserve the dominant shaft-and-base structure but differ in local base geometry and proportions.

These examples complement the quantitative results. The learned embedding appears to capture global geometric structure effectively, while fine-grained dimensional and local geometric differences remain more difficult to discriminate. This behavior provides a plausible explanation for the substantial gap between R@1 and R@5/R@10: even when the exact target is not ranked first, closely related CAD structures frequently occupy the highest-ranked positions.

6 Discussion

The results indicate that contrastive learning can establish a useful correspondence between rendered CAD images and structured parametric CAD command sequences. For canonical queries, the proposed framework achieves 69.60% R@1 and 93.86% R@10, showing that the learned embedding can associate visual geometry with its corresponding parametric construction representation across a gallery of thousands of CAD models.

An important observation is the limited reduction in performance under the mixed canonical/altered query condition. R@1 decreases from 69.60% to 67.40%, while R@10 changes only from 93.86% to 93.72%. This suggests that query modification primarily affects the ordering of the nearest candidates rather than removing the corresponding source CAD representation from the high-similarity

neighborhood. Such behavior is relevant to design reuse, where a modified component may be used to retrieve an existing source model as a starting point for further editing.

The use of sequence-equivalent Recall@K is also important when interpreting these results. Multiple dataset entries can correspond to identical quantized CAD command sequences, so requiring the exact dataset identifier would treat parametrically identical representations as incorrect. At the same time, this criterion captures equivalence only at the command-sequence level and does not account for models that may have similar or identical final geometry but different construction histories.

Finally, the gap between R@1 and R@5/R@10 highlights the value of ranked retrieval. Even when the correct parametric representation is not returned first, it is frequently present among a small number of highly ranked candidates. This makes the framework particularly suitable for interactive CAD search, where users can inspect several retrieved models rather than relying exclusively on the top-ranked result.

A broader implication of these results is that successful cross-modal retrieval does not require the two modalities to describe an object in the same form. The rendered image captures the visible geometry of a design, whereas the parametric command sequence describes how that geometry is constructed through modelling operations and their parameters. Despite this representational gap, the contrastive objective learns a shared space in which corresponding visual and parametric representations remain close. This suggests that structured symbolic representations can provide effective retrieval targets when they encode information that is strongly related to the content observed in the visual modality.

6.1 Limitations

The current evaluation is limited to the parametric command representation and rendered images provided through the ABC/GenCAD data pipeline (Koch et al., 2019; Alam & Ahmed, 2025). Generalization to other CAD systems, modelling conventions, or visual domains such as hand-drawn sketches, technical drawings, and photographs therefore remains to be established.

Sequence-equivalent Recall@K measures equality of the quantized command sequences rather than geometric equivalence between differently constructed models. Finally, although we provide contextual comparisons with prior CAD retrieval methods, differences in query modality, gallery construction, and relevance criteria prevent a fully controlled comparison. Evaluating competing retrieval approaches under the same split, gallery, and correctness criterion remains an important direction for future work.

7 Conclusion

We presented a contrastive image-to-CAD retrieval framework that aligns rendered visual queries with parametric CAD command sequences in a shared embedding space. The model combines a pretrained CLIP RN50 visual encoder with a Transformer-based CAD sequence encoder and is trained using a symmetric contrastive objective.

On the held-out test set, the framework achieves 69.60% R@1, 89.93% R@5, and 93.86% R@10 for canonical queries over a gallery of 7,629 CAD models. When the query set includes both canonical and geometrically altered variants, performance remains close at 67.40% R@1, 89.54% R@5, and 93.72% R@10. These results show that visual CAD representations can be aligned effectively with structured parametric construction sequences and that the source CAD model often remains highly ranked even when the query design is modified.

The results establish that parametric CAD command sequences can serve as effective retrieval targets for visual queries without reducing the CAD model to an image-based representation. More broadly, our findings show that contrastive learning can successfully bridge a visual representation and a structured symbolic representation of the same underlying object. In our setting, the parametric CAD sequence provides a compact description of the modelling operations and parameters that generate the observed geometry. Aligning these representations enables retrieval of reusable and editable CAD models directly from visual input. It also demonstrates the potential of contrastive learning beyond conventional image-text alignment when visual observations can be paired with structured descriptions of the underlying content.

8 AI USE STATEMENT

Generative AI tools were used to assist with literature research, language editing, restructuring of manuscript text, and limited support during code development and debugging. AI-assisted literature research was used to identify potentially relevant publications and research directions; the cited sources and their relevance to this work were subsequently reviewed by the authors. Generative AI was also used to assist in creating the main architecture diagram based on the authors' specification of the proposed framework. All methodological decisions, experimental design, implementation, evaluation, interpretation of results, and final manuscript content were reviewed and verified by the authors. The authors take full responsibility for the accuracy and integrity of the work.

REFERENCES

Md Ferdous Alam and Faez Ahmed. GenCAD: Image-conditioned computer-aided design generation with transformer-based contrastive representation and diffusion priors. *Transactions on Machine Learning Research*, 2025.

A. Ananthakrishnan, Anush Bharathi, V. Dharanivendhan, and Ramanathan Muthuganapathy. CAD-RAG: A multi-modal retrieval augmented framework for user editable 3d CAD model generation. *Journal of WSCG*, 33(1–2):99–110, 2025.

Anush Bharathi, A. Ananthakrishnan, and Ramanathan Muthuganapathy. MM-CAD: A multi-modal CAD dataset and benchmark for cross-modal geometric learning. *Computer Graphics Forum*, 45 (5), 2026.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 1597–1607, 2020.

Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. ABC: A big CAD model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9601–9611, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Swapnil Nagnath Mahajan, Karthik Krishna M., and Ramanathan Muthuganapathy. OrthoCAD-322K: A cross-modal approach for retrieving 3d CAD models from orthographic views using a graph-based framework on a developed large-scale dataset. *Computers & Graphics*, 132:104357, 2025.

Bharadwaj Manda, Shubham Dhayarkar, Sai Mitheran, V. K. Viekash, and Ramanathan Muthuganapathy. CADSketchNet: An annotated sketch dataset for 3d CAD model retrieval with deep neural networks. *Computers & Graphics*, 99:100–113, 2021.

Honghu Pan, Zibo Du, Daxiang Liu, Chengliang Liu, and Xiaoling Luo. Text-to-CAD retrieval: A strong baseline. *arXiv preprint arXiv:2605.05572*, 2026.

Feiwei Qin, Shi Qiu, Shuming Gao, and Jing Bai. 3d CAD model retrieval based on sketch and unsupervised variational autoencoder. *Advanced Engineering Informatics*, 51:101427, 2022.

Feiwei Qin, Liangzhe Zhu, Zijian Xu, Meie Fang, and Ping Li. CADGCL: Unsupervised retrieval of CAD models via boundary representations. *The Visual Computer*, 41(9):6493–6505, 2025.

Yuhan Quan, Huan Zhao, Jinfeng Yi, and Yuqiang Chen. Self-supervised graph neural network for mechanical CAD retrieval. *arXiv preprint arXiv:2406.08863*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763, 2021.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 945–953, 2015.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017.

Rundi Wu, Chang Xiao, and Changxi Zheng. DeepCAD: A deep generative network for computer-aided design models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6772–6782, 2021.

Nomi Yu, Md Ferdous Alam, A. John Hart, and Faez Ahmed. GenCAD-3D: Computer-aided design program generation using multimodal latent space alignment and synthetic dataset balancing. *Journal of Mechanical Design*, 148(3):031703, 2026.