PASTA: Part-Aware Sketch-to-3D Shape Generation with Text-Aligned Prior

Supplementary Material

Overview

The supplementary material provides a detailed description of our proposed method, PASTA, for sketch-to-3D shape generation. It begins with the experimental setup in Section 1, which outlines the data sets [3, 14, 21, 26] used for training and evaluation, the details of the implementation, and the evaluation metrics used to assess the quality of the 3D shapes generated. In Section 2, we elaborate details on the Text-Visual Transformer Decoder, which plays a crucial role in integrating text and visual information to enhance the semantic understanding of input hand-drawn sketches. In Section 3, we analyze Integrated Structure-Graph Network (ISG-Net), a graph-based refinement module designed to improve the structural coherence of the generated 3D models. Finally, in Section 4, we present more qualitative results, demonstrating the effectiveness of PASTA in generating high-quality 3D shapes that faithfully preserve the intended design elements from sketches.

1. Experimental Setup

1.1. Dataset

To evaluate the effectiveness of our proposed method, we utilize multiple datasets [3, 14, 21, 26] representing diverse sketching styles and their corresponding 3D shapes. Specifically, the datasets used in this work include CLIPasso [21] and non-photo-realistic renderings [3], both of which provide diverse stylistic representations of objects. CLIPasso generates abstract, highly simplified depictions, while non-photo-realistic renderings produce more realistic sketches compared to CLIPasso. For evaluation, we used the AmateurSketch-3D [14] and ProSketch-3D [26] datasets. AmateurSketch-3D consists of freehand sketches drawn by non-experts, often exhibiting variability in proportions and details, whereas ProSketch-3D comprises highly detailed, expert-drawn sketches that adhere closely to object structures. Each dataset contributes a distinct perspective on object depiction, spanning a spectrum from abstract simplifications to highly refined artistic renderings. Fig. 1 presents sample sketches from these datasets [3, 14, 21, 26], highlighting differences in abstraction level, detail, and style of chair sketches.

In addition to these datasets [3, 21], we convert realistic images using ControlNet [24] to further analyze our approach on real-world photo based 3D shape generation. Specifically, we apply ControlNet to sketches from CLIPasso and non-photo-realistic renderings, producing enhanced versions that preserve the original structural essence of the sketches while incorporating greater realism. Using ControlNet to guide image generation with fine-grained control, we ensure that the generated results retain the fundamental characteristics of the input sketches while resembling realistic RGB images. This allows us to bridge the gap between abstract grayscale representations and photo-realistic RGB images, offering a richer set of inputs for 3D generation. The generated results are displayed in Fig. 2, demonstrating how ControlNet enhances the visual fidelity of sketches while maintaining their structural shape through examples in the realistic RGB images.

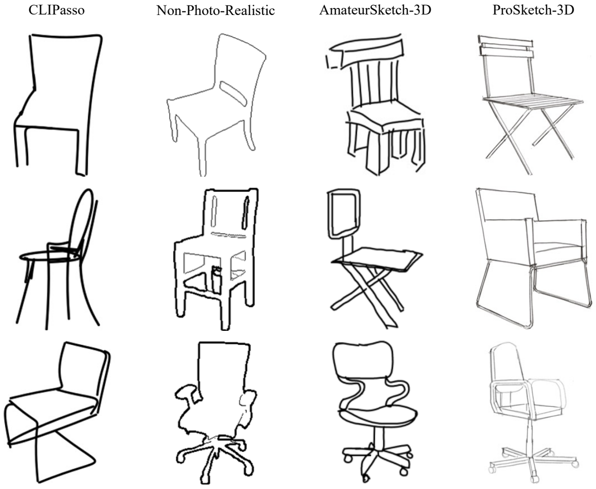


Figure 1. Sample images from the datasets used in our experiments, including CLIPasso [21], non-photo-realistic renderings [3], AmateurSketch-3D [14], and ProSketch-3D [26]. These datasets capture a range of sketching styles, from highly abstract representations to detailed, expert-drawn sketches.

1.2. Implementation Details

Training Configuration. For the training and evaluation of our model, we adopt a carefully designed experimental setup to ensure robust performance and fair comparisons with existing methods. Our model is trained on a single RTX 3090 GPU for approximately 38 hours for the chair category. The training process follows a batch size of 16 and employs an initial learning rate of 10^{-4}, which is dynamically adjusted using the OneCycle learning rate scheduler [18] to facilitate stable convergence. This scheduler gradually increases the learning rate in the early stages of training before decaying it towards the end, preventing premature convergence and improving generalization. The model is optimized using the Adam optimizer, and we train for 650 epochs, ensuring sufficient iterations for convergence while mitigating the risk of overfitting.

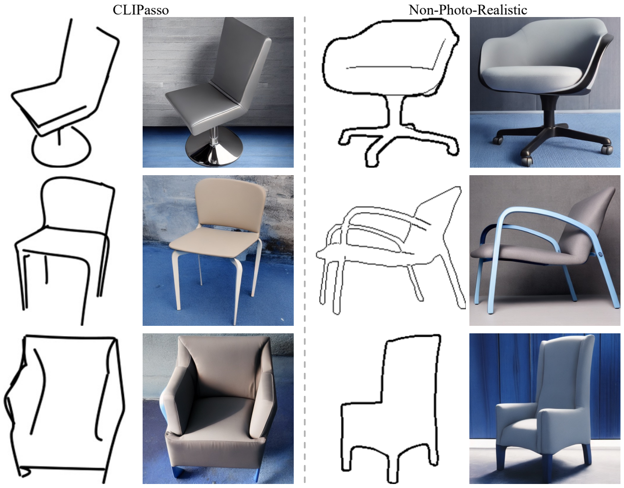


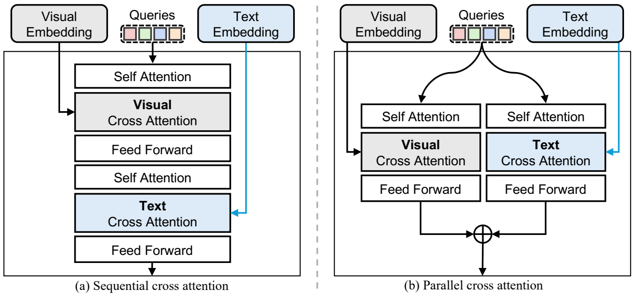
Figure 2. Illustration of sketches processed using ControlNet [24]. The transformed sketches retain the structural essence of the originals while incorporating enhanced realism.

Architectural Specifications. In addition to these experimental settings, we provide detailed specifications regarding the architectural components of PASTA. We set the number of queries and Gaussian mixture models to $N = 16$. Within the PartGCN module, the number of graph clustering groups is set to $K = 4$, ensuring effective part-wise feature aggregation. The overall model architecture is designed with two layers in the graph-convolution network (GCN), allowing for a balance between expressiveness and computational efficiency. As described in Equation (9) of the main paper, the weighting factor α in IndivGCN and PartGCN is set to 0.8 based on experimental results. Furthermore, the loss weights for different components, including λ_{align}, λ_{indiv}, and λ_{part}, are set to 1.0, 0.1, and 0.1, respectively, following the formulation in Equation (11) of the main paper, ensuring optimal performance in various sketch styles and object categories.

1.3. Evaluation Metrics

For evaluation, we employ three metrics to assess the quality of the 3D shapes generated. Chamfer distance (CD) quantifies the accuracy of the point-wise reconstruction by measuring the discrepancy between the predicted point clouds set P and the ground-truth point clouds set G:

$$\text{CD}(P,G) = \frac{1}{|P|}\sum_{p\in P}\min_{g\in G}\|p-g\|_2^2 + \frac{1}{|G|}\sum_{g\in G}\min_{p\in P}\|p-g\|_2^2. \tag{1}$$



(a) Sequential cross attention

(b) Parallel cross attention

Figure 3. Comparison of two architectures for the Text-Visual Transformer Decoder: (a) sequential cross-attention and (b) parallel cross-attention mechanisms.

	AmateurSketch-3D			ProSketch-3D		
Methods	CD ↓	EMD ↓	FID ↓	CD ↓	EMD ↓	FID ↓
(a) Sequential	**0.090**	**0.071**	**143.9**	**0.055**	**0.049**	**112.2**
(b) Parallel	0.095	0.074	150.2	0.071	0.061	120.7

Table 1. Quantitative comparison of the two Text-Visual Transformer Decoder architectures. (a) Sequential method consistently outperforms (b) parallel method across multiple evaluation metrics, highlighting the benefits of first enriching visual embeddings before merging them with textual information.

The Earth Mover's Distance (EMD) quantifies structural differences by determining the minimum cost required to transform one point cloud into another, based on an optimal correspondence $\pi \in \Pi(P,G)$. Here, $\Pi(P,G) \in \mathbb{R}^{n\times m}$ consists of elements in the range between 0 and 1, such that the sum of each row and each column equals one.

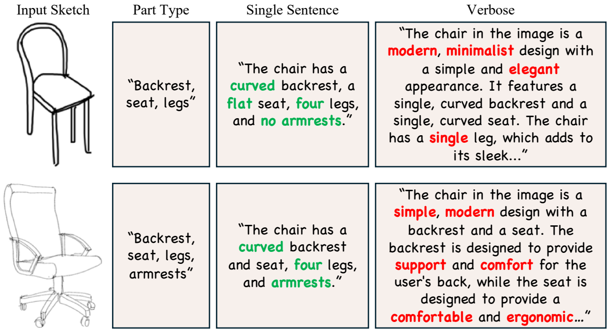
$$\text{EMD}(P,G) = \min_{\pi\in\Pi(P,G)} \sum_{i=1}^{n}\sum_{j=1}^{m} \pi_{i,j}\|p_i - g_j\|. \tag{2}$$

For the calculation of CD and EMD, 2,048 points are sampled from both the ground truth mesh and the generated mesh. Finally, the Fréchet Inception Distance (FID) [6] is used to assess the realism of the generated shapes by comparing the feature distributions of the rendered 3D models with those of real-world reference data:

$$\text{FID} = \frac{1}{20}\sum_{i=1}^{20}\left(\|\mu_i - \mu_i'\|_2^2 + \text{Tr}\left(\Sigma_i + \Sigma_i' - 2\sqrt{\Sigma_i\Sigma_i'}\right)\right). \tag{3}$$

To compute the FID score, we first sample 20 different views and render both the ground truth shape S the generated shape S'. The features are then extracted from these images using the Inception-V3 network [19], which maps each image to a probability distribution across 1,000 classes. From this distribution, we compute the mean μ_i and the covariance matrix Σ_i for each image i. These statistics are then used to calculate the final FID score. To improve the accuracy of the calculation, we employ the shading-image-based FID metric.

By systematically integrating these experimental configurations, our approach is designed to generate high-fidelity 3D shape generations while maintaining structural consistency across different types of input sketches. The following sections further analyze our results, highlighting details in qualitative and quantitative improvements.



Input Sketch	Part Type	Single Sentence	Verbose
	“Backrest, seat, legs”	“The chair has a **curved** backrest, a **flat** seat, **four** legs, and **no armrests**.”	“The chair in the image is a **modern**, **minimalist** design with a simple and **elegant** appearance. It features a single, curved backrest and a single, curved seat. The chair has a **single** leg, which adds to its sleek…”
	“Backrest, seat, legs, armrests”	“The chair has a **curved** backrest and seat, **four** legs, and **armrests**.”	“The chair in the image is a **simple**, **modern** design with a backrest and a seat. The backrest is designed to provide **support** and **comfort** for the user's back, while the seat is designed to provide a **comfortable** and **ergonomic**…”

Figure 4. Examples of different text description styles used in the VLM [12]. The three types include Part Type, Single Sentence, and Verbose descriptions.

	AmateurSketch-3D			ProSketch-3D		
Methods	CD ↓	EMD ↓	FID ↓	CD ↓	EMD ↓	FID ↓
CLIP [15]	0.101	0.080	152.1	0.066	0.053	117.7
BLIP2 T5 [11]	0.095	0.073	147.4	0.061	0.051	115.6
LLaVa-13B [12]	0.091	**0.071**	144.3	**0.054**	0.050	113.9
LLaVa-7B [12]	**0.090**	**0.071**	**143.9**	0.055	**0.049**	**112.2**

Table 2. Performance comparison of different VLMs [11, 12, 15] in our framework. LLaVa-7B [12] achieves the best balance between CD, EMD, and FID that making it the most suitable choice for enhancing text-guided 3D shape generation.

2. Details of Text-Visual Transformer Decoder

The Text-Visual Transformer Decoder is a key contribution in our framework, responsible for integrating text priors with visual features to enhance 3D shape generation from sketches. In this section, we present the architectural design of the decoder, analyze the impact of different vision-language models (VLMs) [11, 12, 15], and evaluate the role of input text descriptions in guiding the generation process.

2.1. Architectural Design of Text-Visual Transformer Decoder

Fig. 3 presents two different architectures to integrate text and visual condition within our Text-Visual Transformer Decoder. In method (a), the visual embedding is first processed through a cross-attention mechanism before being fused with text embeddings via a second cross-attention operation. In contrast, method (b) applies cross-attention separately to both visual and text embeddings before merging them later in the process. Tab. 1 compares the performance of these two approaches, showing that method (a) consistently outperforms method (b). The key advantage of method (a) is that the initial cross-attention incorporates visual condition into the query, allowing the subsequent text cross-attention to extract more relevant information. Since the text embeddings extracted from the VLM [12] are aligned with the visual latent space, integrating the sketch embedding into the query first aids in the consolidation of corresponding part-specific information. In contrast, as in method (b), applying cross-attention between a query devoid of visual cues and the text embeddings may lead to unintended information exchange. This results in improved alignment between sketches and their corresponding 3D shapes, leading us to adopt method (a) in our framework.

2.2. Impact of Vision-Language Models

To assess the influence of different VLMs [11, 12, 15] on our framework, we conduct a comparative analysis using various VLMs, as summarized in Tab. 2. When employing CLIP [16], only marginal performance improvements are observed, likely due to its categorical training and alignment, which are specifically optimized for classification tasks. In contrast, models that excel in image captioning and visual question answering (VQA) yield more substantial enhancements. Notably, BLIP2 [11], lacking explicit fine-tuning on instruction data, frequently produces suboptimal outcomes compared to LLaVA-7B/13B. For instance, it sometimes generates responses that are irrelevant to the user's instructions (e.g., *“a chair is a piece of furniture with a seat and backrest.”*) or omits critical components, such as the number of legs and the presence of armrests (e.g., *“a chair with a backrest and seat that are shaped in the form of a horseshoe.”*). In contrast, the LLaVA-7B model consistently provides responses that comprehensively incorporate the necessary elements. While LLaVA-13B delivers descriptions similar to those of the 7B model, its performance is marginally inferior. Considering model size, computational cost, and overall performance balance, we therefore adopt the 7B model in this work.

Effect of Description Types on Text Embeddings. The quality and structure of text descriptions play a valuable role in guiding the 3D generation process. Fig. 4 provides examples of different description styles, corresponding to part type based descriptions, single sentence descriptions, and verbose descriptions. For instance, in the **Verbose**, the description of such discrepancies arises due to overemphasis on stylistic attributes like “modern”, “minimalist”, and “elegant” which can mislead the 3D generation model. On the other hand, **Part Type**, such as “Backrest, seat, legs, armrests,” provides a fundamental structural understanding. However, they lack details about the shape and number of components, which are crucial for precise shape generation. In contrast, **Single Sentence** strikes a balance between clarity and informativeness. They specify the number and form of object components while avoiding unnecessary complexity. Our analysis, supported by Table 5 in the main paper, reveals that verbose descriptions often introduce hallucinations and excessive modifiers leading to inconsistencies in shape generation. As a result, our framework adopts single sentence descriptions to maximize the accuracy and reliability of text-guided 3D shape generation, by maintaining specificity while avoiding hallucinations.

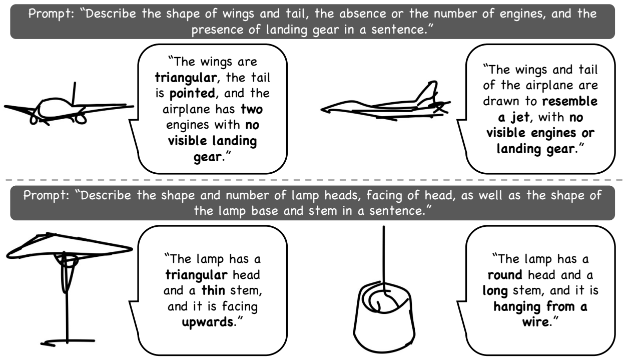

Prompt: "Describe the shape of wings and tail, the absence or the number of engines, and the presence of landing gear in a sentence."
"The wings are triangular, the tail is pointed, and the airplane has two engines with no visible landing gear."
"The wings and tail of the airplane are drawn to resemble a jet, with no visible engines or landing gear."
Prompt: "Describe the shape and number of lamp heads, facing of head, as well as the shape of the lamp base and stem in a sentence."
"The lamp has a triangular head and a thin stem, and it is facing upwards."
"The lamp has a round head and a long stem, and it is hanging from a wire."

Figure 5. This figure presents prompts used for airplane and lamp sketches, along with the text descriptions produced by the VLM [12]. The generated descriptions capture the key structural features of each sketch, providing valuable semantic information.

Text-Based Descriptions and Generated Outputs. We extend our analysis beyond the chair category to include airplane and lamp, as shown in Fig. 5. The results demonstrate that VLM [12] effectively captures fine-grained structural details across different object types, showcasing its ability to perform effectively across diverse shape categories. By utilizing text-aligned priors, PASTA successfully distinguishes between distinct design elements, such as the wing and fuselage in airplanes or the lamp head and base in lamps. These findings highlight the robustness of text-visual integration strategy, confirming its effectiveness in enhancing the semantic understanding of sketches across multiple categories.

3. Analysis of ISG-Net

The Integrated Structure-Graph Network (ISG-Net) is designed to refine the structural consistency of 3D shape generation by incorporating graph-based reasoning. The network consists of two key modules: IndivGCN, which focuses on fine-grained feature extraction, and PartGCN, which aggregates and refines part-level information. In this section, we first explore related work on Graph Neural Networks (GNNs) and then analyze the effects of the clustering method used in PartGCN, the impact of the number of clusters K, and the balance between IndivGCN and PartGCN using the parameter α.

3.1. Releated Work on Graph Neural Networks

Graph neural networks (GNNs) [17] model complex structured data as nodes and edges, thus effectively learning relationships within the data. Early work focused on message-passing mechanisms to aggregate information among nodes, and later convolutional neural networks (CNNs) were generalized to graph-structured data, leading to the development of graph convolutional networks (GCNs) that facilitate information exchange between neighboring nodes. GCNs can be classified into spatial-based methods [2, 13], which apply trainable filters directly to connected nodes, and spectral-based methods [5, 9], which define locality through spectral analysis. Such graph-based neural networks have been applied not only to social networks [4] but also to various computer vision tasks [1, 7, 8, 10, 20, 22, 23]. In this work, we apply GCNs to Gaussian mixture models (GMMs) for 3D shape representation. We define the graph structure by producing an adjacency matrix based on the distances between Gaussians in 3D space, thereby enabling information exchange between adjacent nodes for more structurally accurate shape generation and reconstruction. Furthermore, by grouping similar nodes and performing additional graph operations, we achieve robust structural learning.

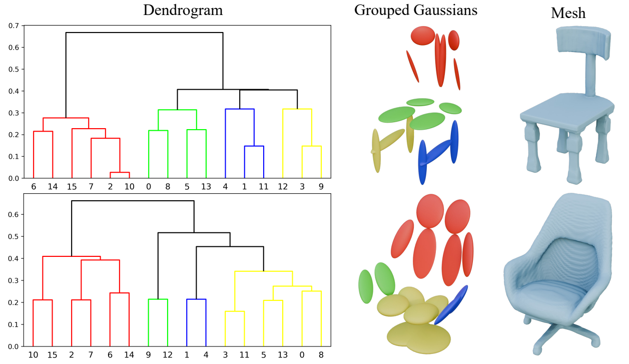


Figure 6. Dendrogram [25] and visualization of Gaussians with the mesh illustrating the clustering process in PartGCN, where $N = 16$ nodes are grouped into $K = 4$ clusters to enhance part-wise feature aggregation.

3.2. Clustering Method for PartGCN

To perform PartGCN operations, N queries and Gaussians must be assigned to K subsets. To achieve this, we employ a hierarchical clustering technique [25] to partition the parts. This clustering is applied to both the pseudo-ground truth and the predicted queries. For pseudo-ground truth, clustering is performed based on $\mathbf{A}_I \in \mathbb{R}^{N \times N}$ to form K groups, and the coordinates of the Gaussians within each group are aggregated using average pooling to obtain a representative coordinate. The distances between these coordinates are then computed to produce the final pseudo-ground truth adjacency matrix $\mathbf{A}_P \in \mathbb{R}^{K \times K}$. Simultaneously, predicted queries are pooled for each group to obtain $\mathbf{Q}_P \in \mathbb{R}^{K \times d}$, which is processed through f_{part} and subsequently subjected to an dot product operation to produce the predicted adjacency matrix $\tilde{\mathbf{A}}_P \in \mathbb{R}^{K \times K}$. As illustrated in Fig. 6, the clustering process is depicted through a dendrogram [25], where N nodes are grouped into K clusters. In particular, it flexibly groups dynamically varying Gaussians (*e.g.*, those located near the legs versus those near the armrests). This method enables PartGCN to accurately identify and select relevant structural parts, ensuring that the network focuses efficiently on the details of the part level while maintaining overall structural integrity.

	AmateurSketch-3D		
Methods	CD ↓	EMD ↓	FID ↓
K = 2	0.098	0.081	160.2
K = 4	**0.090**	**0.071**	**143.9**
K = 6	0.093	0.076	149.0
K = 8	0.094	0.078	156.4

Table 3. Performance comparison for different cluster numbers K. Too few clusters weaken structural representation, while too many reduce synergy with IndivGCN.

3.3. Effect of Cluster Number K

The experimental results on varying the number of clusters K reveal important insights into the impact of cluster selection on the performance of our model. When K is too small, there is insufficient granularity in the clustering process, causing PartGCN to lose critical structural relationships. As a result, the overall structural integrity of the output suffers and the model does not adequately represent the coherence of the object. On the other hand, when K is excessively large, the model faces a different challenge. Although more clusters may provide a higher resolution of detail, it also introduces redundancy and excessive complexity. In this case, PartGCN becomes overloaded with unnecessary information, leading to a loss of synergy with IndivGCN. This imbalance occurs because excessive detail hampers the effective integration of local features, causing IndivGCN and PartGCN to work against each other rather than complement each other. The model's performance, therefore, suffers from a lack of coherence between the fine-grained details and the global structure. As shown in Tab. 3, K=4 ensures that both components work harmoniously, leading to the generation of high-quality 3D shapes without sacrificing performance or structural integrity. This process allows PartGCN to contribute effectively to the overall structure while maintaining the balance with IndivGCN.

3.4. Influence of α in IndivGCN and PartGCN

In our experiments, we use the parameter α as a weighting coefficient for the outputs of IndivGCN and PartGCN, as described in Equation (9) of the main paper. Specifically, α scales the output of IndivGCN, which captures fine-grained local features, while the complement $(1-\alpha)$ scales the output of PartGCN, which encodes global structural elements. The experimental results, shown in Tab. 4, demonstrate that the best performance is achieved when α is set to 0.8, striking the optimal synergy between these two components. In this configuration, IndivGCN is weighted more heavily, which allows it to preserve fine-grained local details that are crucial for the accurate representation of intricate features. At the same time, PartGCN is still capable of effectively contributing to the integrity of the global structure by encoding broader spatial relationships and consistency at the part level, as it is scaled by $(1 - \alpha)$. This balance is key to ISG-Net success, while IndivGCN focuses on refining local geometry and intricate details, PartGCN provides essential structural support, preventing shape distortions, and preserving overall coherence. Consequently, this leads to the best overall performance of ISG-Net, enabling it to generate 3D shapes that are rich in fine-grained detail and structurally sound, preserving oversimplified sketches.

	AmateurSketch-3D		
Methods	CD ↓	EMD ↓	FID ↓
$\alpha = 0.0$	0.095	0.084	157.0
$\alpha = 0.2$	0.095	0.081	155.4
$\alpha = 0.4$	0.092	0.077	151.8
$\alpha = 0.6$	0.094	0.077	152.1
$\alpha = 0.8$	**0.090**	**0.071**	**143.9**
$\alpha = 1.0$	0.092	**0.071**	145.3

Table 4. Effect of α on IndivGCN and PartGCN balance. The best performance is achieved at $\alpha = 0.8$, optimizing detail and structure integration.

4. Qualitative Results

Additional qualitative results across different object categories, including chair, airplane, and lamp, are provided to further support our experimental findings. For chairs (Fig.7, 8), our model preserves key structures such as leg details, backrest curvature, and armrests, even in abstract sketches. For airplanes (Fig.9), PASTA generates coherent components like wings, fuselage, engines, and tail. For lamps (also Fig. 9), it captures fine details such as frame structures, lampshades, and support bases. These results highlight strong generalization and ability to handle diverse and complex shapes with high structural fidelity of PASTA. To further validate these qualitative observations through human perception, we conducted a user study. We randomly selected 50 objects (30 chairs, 10 airplanes, and 10 lamps) and chose 5 examples for editing from our entire set of outputs. Then, we recruited 100 participants via Amazon Mechanical Turk (AMT) and presented them with two types of rendered-image-based questions: (i) a preference comparison between PASTA and SENS based on performance, and (ii) a visual quality rating using a 7-point Likert scale. The results show in Tab. 5, across all categories and editing cases, our method significantly outperforms SENS within a 95% confidence interval, further reinforcing the effectiveness and visual quality of our approach.

	Chair		Airplane		Lamp		Editing	
Method	Prefer (%)	Likert ↑	Prefer (%)	Likert ↑	Prefer (%)	Likert ↑	Prefer (%)	Likert ↑
SENS	7.63±1.11	4.70±0.09	18.70±1.24	5.16±0.23	14.00±1.83	4.29±0.24	14.60±3.84	4.04±0.26
PASTA	**92.37±1.11**	**5.95±0.02**	**81.30±1.24**	**5.59±0.20**	**86.00±1.83**	**5.96±0.16**	**85.40±3.84**	**6.17±0.17**

Table 5. User study results: preference (%) and average Likert ratings for each category. Higher is better.

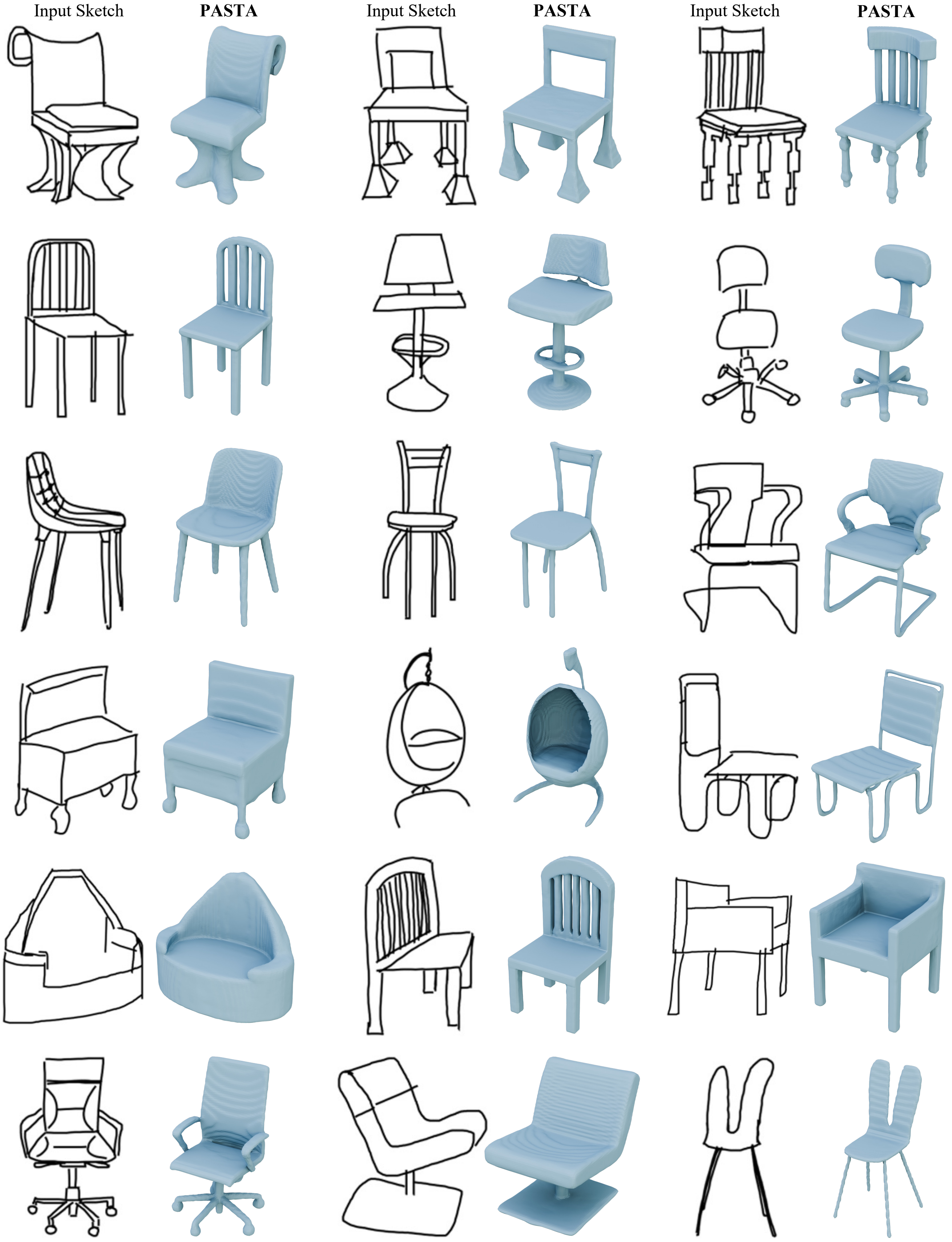


Figure 7. Qualitative results for chair reconstructions, demonstrating the preservation of key structural elements such as leg orientation, backrest curvature, and armrest presence.

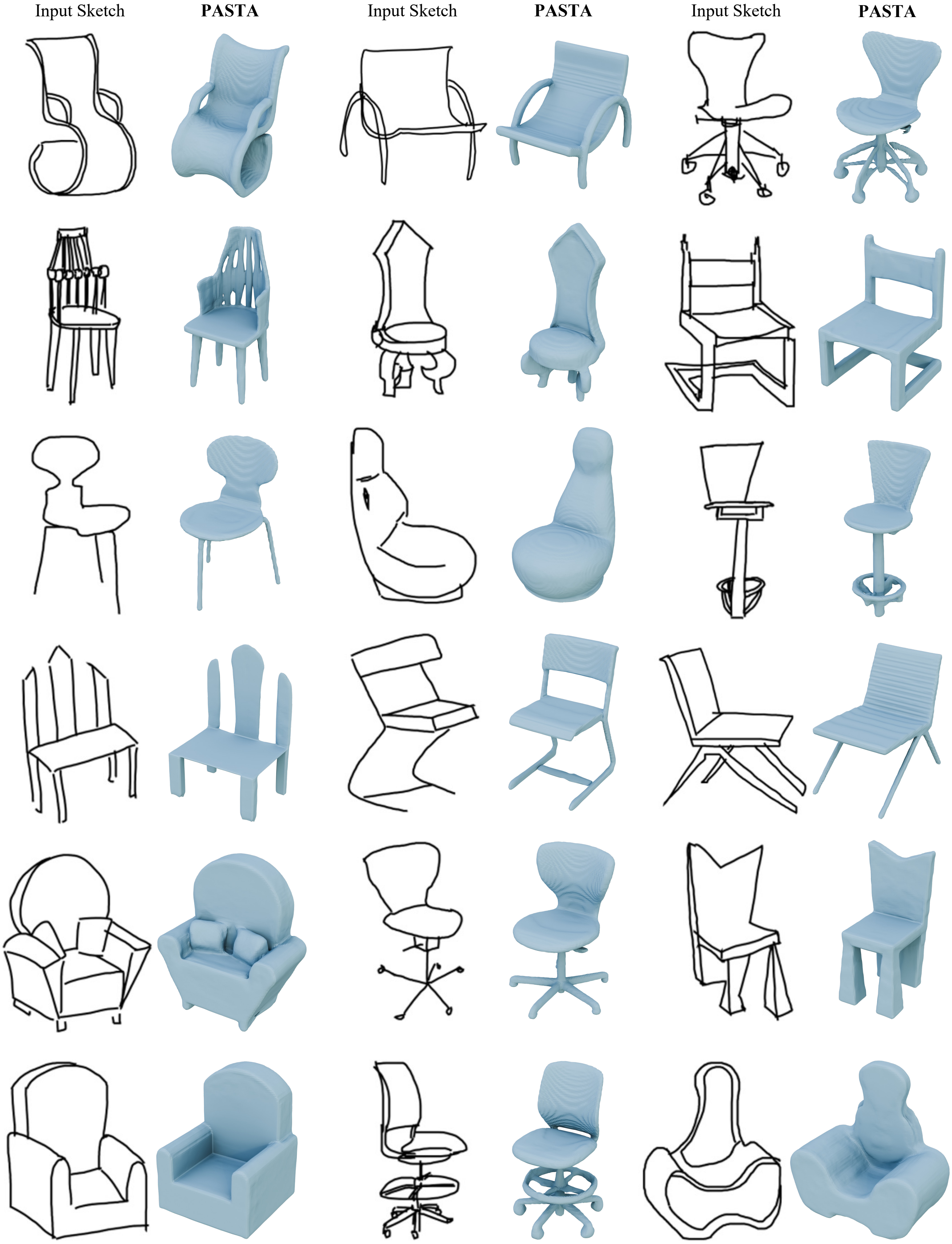


Figure 8. Qualitative results for chair reconstructions, demonstrating the preservation of key structural elements such as leg orientation, backrest curvature, and armrest presence.

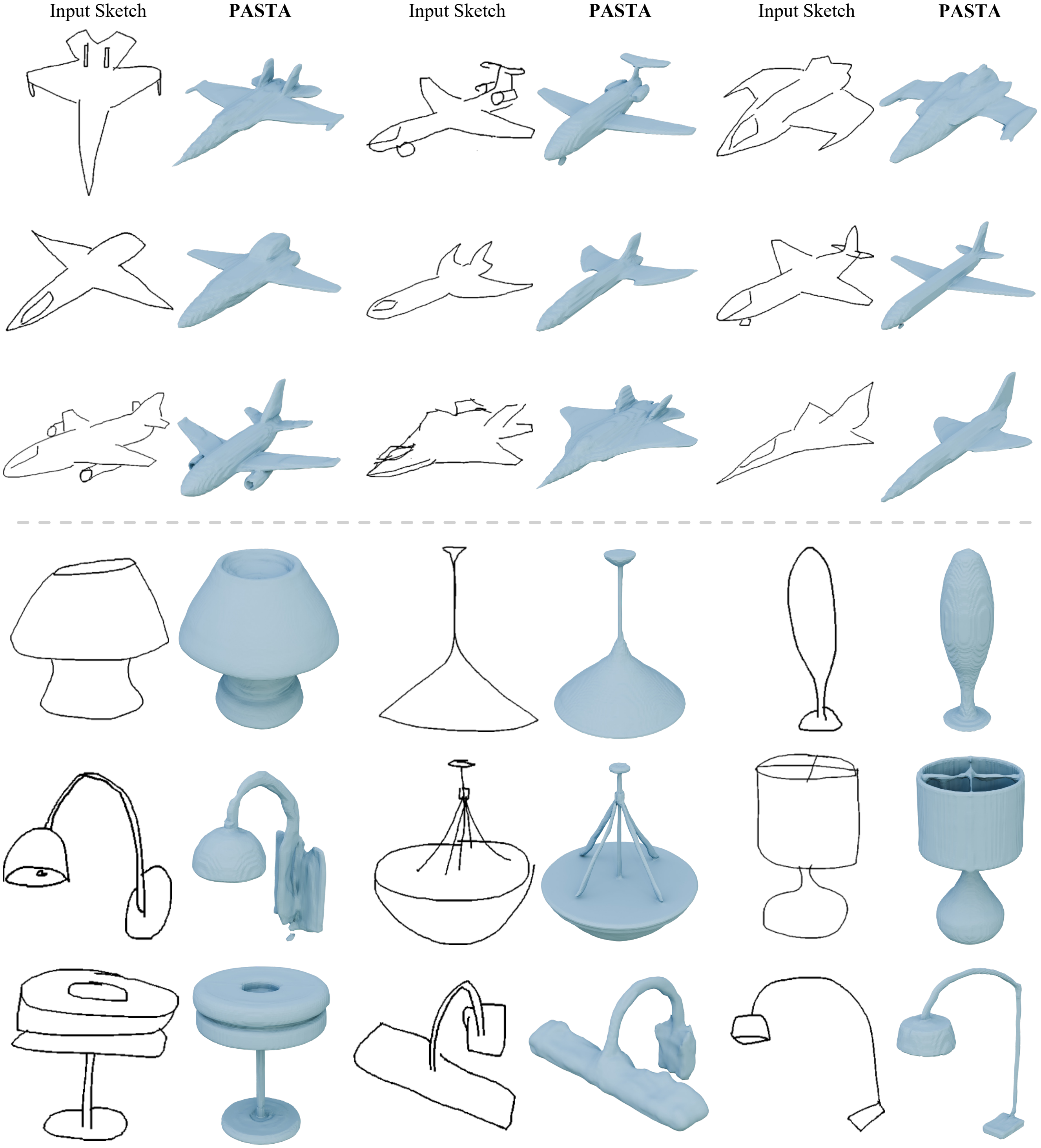


Figure 9. Additional qualitative results for airplanes and lamps, showcasing accurate reconstruction of aircraft components and intricate lamp structures.

References

[1] Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Learning pedestrian group representations for multi-modal trajectory prediction. In *European Conference on Computer Vision*, pages 270–289. Springer, 2022. 4

[2] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013. 4

[3] Caroline Chan, Frédo Durand, and Phillip Isola. Learning to generate line drawings that convey geometry and semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7915–7925, 2022. 1

[4] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017. 4

[5] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015. 4

[6] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 2

[7] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018. 4

[8] Sungjune Kim, Hadam Baek, Seunggwan Lee, Hyung-gun Chi, Hyerin Lim, Jinkyu Kim, and Sangpil Kim. Enhanced motion forecasting with visual relation reasoning. In *European Conference on Computer Vision*, pages 311–328. Springer, 2024. 4

[9] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 4

[10] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4558–4567, 2018. 4

[11] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 3

[12] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 3, 4

[13] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In *International conference on machine learning*, pages 2014–2023. PMLR, 2016. 4

[14] Anran Qi, Yulia Gryaditskaya, Jifei Song, Yongxin Yang, Yonggang Qi, Timothy M Hospedales, Tao Xiang, and Yi-Zhe Song. Toward fine-grained sketch-based 3d shape retrieval. *IEEE transactions on image processing*, 30:8595–8606, 2021. 1

[15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3

[16] Aditya Sanghi, Rao Fu, Vivian Liu, Karl DD Willis, Hooman Shayani, Amir H Khasahmadi, Srinath Sridhar, and Daniel Ritchie. Clip-sculptor: Zero-shot generation of high-fidelity and diverse shapes from natural language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18339–18348, 2023. 3

[17] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1): 61–80, 2008. 4

[18] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, pages 369–386. SPIE, 2019. 1

[19] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 2

[20] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2017. 4

[21] Yael Vinker, Ehsan Pajouheshgar, Jessica Y Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *ACM Transactions on Graphics (TOG)*, 41(4):1–11, 2022. 1

[22] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Learning combinatorial embedding networks for deep graph matching. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3056–3065, 2019. 4

[23] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017. 4

[24] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 1, 2

[25] Ying Zhao, George Karypis, and Usama Fayyad. Hierarchical clustering algorithms for document datasets. *Data mining and knowledge discovery*, 10:141–168, 2005. 4

[26] Yue Zhong, Yonggang Qi, Yulia Gryaditskaya, Honggang Zhang, and Yi-Zhe Song. Towards practical sketch-based 3d shape generation: The role of professional sketches. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(9):3518–3528, 2020. 1