Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.068500

ARTICLE

PolyDiffusion: A Multi-Objective Optimized Contour-to-Image Diffusion Framework

Yuzhen Liu[1,2], Jiasheng Yin[1,2], Yixuan Chen[1,2], Jin Wang[1,2], Xiaolan Zhou[1,2] and Xiaoliang Wang[1,2,*]

[1]School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan, 411201, China
[2]Sanya Research Institute, Hunan University of Science and Technology, Sanya, 572024, China

*Corresponding Author: Xiaoliang Wang. Email: fengwxl@hnust.edu.cn

Received: 30 May 2025; Accepted: 07 August 2025; Published: 23 September 2025

ABSTRACT: Multi-instance image generation remains a challenging task in the field of computer vision. While existing diffusion models demonstrate impressive fidelity in image generation, they often struggle with precisely controlling each object's shape, pose, and size. Methods like layout-to-image and mask-to-image provide spatial guidance but frequently suffer from object shape distortion, overlaps, and poor consistency, particularly in complex scenes with multiple objects. To address these issues, we introduce PolyDiffusion, a contour-based diffusion framework that encodes each object's contour as a boundary-coordinate sequence, decoupling object shapes and positions. This approach allows for better control over object geometry and spatial positioning, which is critical for achieving high-quality multi-instance generation. We formulate the training process as a multi-objective optimization problem, balancing three key objectives: a denoising diffusion loss to maintain overall image fidelity, a cross-attention contour alignment loss to ensure precise shape adherence, and a reward-guided denoising objective that minimizes the Fréchet distance to real images. In addition, the Object Space-Aware Attention module fuses contour tokens with visual features, while a prior-guided fusion mechanism utilizes inter-object spatial relationships and class semantics to enhance consistency across multiple objects. Experimental results on benchmark datasets such as COCO-Stuff and VOC-2012 demonstrate that PolyDiffusion significantly outperforms existing layout-to-image and mask-to-image methods, achieving notable improvements in both image quality and instance-level segmentation accuracy. The implementation of PolyDiffusion is available at https://github.com/YYYYYJS/PolyDiffusion (accessed on 06 August 2025).

KEYWORDS: Diffusion models; multi-object generation; multi-objective optimization; contour-to-image

1 Introduction

In recent years, diffusion-based generative models [1] have advanced rapidly, achieving state-of-the-art fidelity in image, video, and 3D modeling, gradually superseding traditional GANs [2]. However, text-to-image diffusion approaches [3,4] still struggle with multi-instance scenes: a single textual prompt provides only global guidance, leaving per-object details—shape, pose, and spatial arrangement—underconstrained. As a result, multi-object generation must be formulated as a multi-objective optimization problem, balancing overall image fidelity with individual instance constraints.

Layout-to-image methods [5] address part of this challenge by using bounding-box layouts and class labels to enforce coarse spatial positioning. LayoutDiffusion further improves multimodal fusion between layout and image features, yielding more coherent object placement. However, reliance on high-level layout diagrams still falls short of fine-grained shape control and inter-object consistency in complex

Copyright © 2025 The Authors. Published by Tech Science Press.
This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

scenes. For instance, text-to-image or layout-to-image methods often suffer from issues like object shape distortion, spatial overlap, and semantic inconsistency when generating multiple objects simultaneously. These problems arise because layout diagrams provide only high-level semantic guidance without fine-grained constraints, which limits the generation of realistic multi-object scenes. Specifically, in many cases, objects may overlap in the generated scene, or the shapes of the objects may become distorted due to the lack of precise control over their spatial relationships. Moreover, semantic inconsistencies arise when objects with similar categories or spatial proximity are generated, leading to unrealistic or poorly placed objects.

To illustrate, Contour Wavelet Diffusion [6] introduces a fast image generation method using wavelet transforms, which improves efficiency while maintaining image quality. However, this method still faces challenges when generating multi-object scenes due to the limited control over object layout and interactions. Similarly, Unicontrol [7] is a representative work that supports multiple control signals for controllable visual generation. While it enables flexible multi-task learning, its focus is on broader control, lacking the precise instance-level control required for multi-object generation.

In contrast, we propose PolyDiffusion, a multi-objective optimized contour-to-image diffusion framework that addresses these challenges. By encoding detailed object contours as closed-loop boundary sequences, PolyDiffusion offers precise control over the shape, size, and placement of multiple objects in a scene. This allows for better spatial consistency and object interaction, overcoming the limitations of previous methods that struggle with complex object layouts. Through the introduction of multi-objective optimization, PolyDiffusion refines contour-to-image generation by incorporating both global and local constraints, providing higher fidelity and flexibility in multi-object image generation.

Our approach builds on the foundation of layout-to-image and mask-to-image generation methods, unifying them into a flexible framework. Specifically, the input for layout-to-image generation consists of a set of bounding boxes and class labels for the objects within the scene, while the input for mask-to-image generation is usually text and shape descriptions. We noticed that InstanceDiffusion [8] works very similar to ours. In our task, the mask of each object is described as a polygon sequence, and the layout diagram, as a simpler 2D box, can be easily extended to represent polygon sequences. In contrast, much of the current non-text-to-image generation research is limited to single-object image generation, rather than focusing on generating images with multiple objects. Our goal is to use more flexible control conditions to guide the generation of multi-object images.

The core contributions of PolyDiffusion are as follows:

- Polygon Diffusion Model for Multi-Object Generation: We introduce a polygon diffusion model that builds on the layout-to-image framework, transforming contour maps into image-generating objects. This model decouples multiple objects, enabling the generation of high-quality, diverse images with precise control over the position, size, and shape of each object.
- Spatial Relationship Modeling via Contour Representation: We propose a contour point coordinate representation based on circular relative position encoding, which combines shape and spatial information to better capture an object's geometric features. By placing the contours of different objects within a unified coordinate system, the model enhances the understanding of spatial relationships between objects, improving the overall image quality.
- Unified Control of Global Conditions and Object Details: To facilitate interaction between objects, we introduce CFM modules that enable contextual perception based on object distance and class similarity. Additionally, we achieve seamless contour-image fusion using object-aware and position-sensitive design. A novel alignment loss is introduced to ensure consistency between conditions and generated objects, and further fine-tuning of the final denoising step enhances the quality of the generated images.

Through these innovations, PolyDiffusion sets a new standard for contour-to-image generation, providing precise, flexible control over multi-object scenes and addressing challenges that have limited the effectiveness of previous methods.

2 Related Work

The related work is mainly introduced from the aspects of diffusion models and image generation. Image generation is a critical task in computer vision. A range of works has been proposed, from early GAN series [2,9,10] to VAE series [11] and stream model series.

Recently, advances in image generation based on diffusion models have led to significant progress, with text-to-image generation standing out as one of the most advanced applications. This success is largely due to the availability of large-scale text-image pair datasets, which are much easier to collect compared to other types of data. The development of models like CLIP [12] has enabled diffusion models to achieve remarkable results in this field. While text-image data remains the most abundant, there is also ongoing work exploring other conditions beyond text-to-image, such as layout-to-image [13], pose-to-image, depth map-to-image, and mask-to-image [14–16].

LayoutDiffusion is a controllable diffusion framework designed specifically for layout-to-image generation. By transforming the complex fusion of layout and image data into a unified structural representation, LayoutDiffusion improves both generation quality and controllability. Unlike traditional methods, it offers precise control over object placement and size while maintaining high image quality [17]. Similarly, SmartMask is a context-aware diffusion model developed to enhance object insertion and layout generation. This model eliminates the need for masks, enabling precise object placement, and supports iterative refinement for fine-grained layout generation. By preserving background content and offering better control over object placement, SmartMask further advances semantic-to-image generation [18].

In addition to these methods, Contour Wavelet Diffusion [6] introduces a fast image generation method using wavelet transforms, improving efficiency while maintaining image quality. It accelerates training and inference with batch-normalized stochastic attention and a balanced loss function. In contrast, PolyDiffusion focuses on precise control over object placement and layout in multi-object scenes. While Contour Wavelet Diffusion emphasizes efficiency, PolyDiffusion excels in handling complex object interactions, offering better control and higher fidelity in multi-object image generation.

Unicontrol is one of the most representative works in recent years, proposing a unified diffusion model for controllable image generation. It supports multiple control signals and enables multi-task learning in a zero-shot setting [7]. In contrast, PolyDiffusion focuses on contour-to-image generation, providing precise control over object layout and spatial structure, particularly in multi-object generation, where it offers higher accuracy and flexibility.

While Transformer-based methods, such as Vision Transformers (ViT) [19] and Swin Transformers [20], have demonstrated success in handling long-range dependencies in image generation tasks, PolyDiffusion focuses on leveraging contour-based representations to offer more precise control over object shapes and spatial relationships. This makes our approach particularly well-suited for multi-object generation, where managing object interactions and maintaining spatial consistency are crucial.

Our work lies somewhere between layout-to-image and mask-to-image generation, or our approach somewhat unifies the two. Specifically, the input for layout-to-image generation consists of a set of bounding boxes and class labels for the objects within the scene, while the input for mask-to-image generation is usually text and shape descriptions. We noticed that InstanceDiffusion [8] works very similar to ours. In our task, the mask of each object is described as a polygon sequence, and the layout diagram, as a simpler 2D box,

can be easily extended to represent polygon sequences. In contrast, much of the current non-text-to-image generation research is limited to single-object image generation, rather than focusing on generating images with multiple objects. Our goal is to use more flexible control conditions to guide the generation of multi-object images.

3 Method

In this section, we present our Poly Diffusion, as shown in Fig. 1. The whole framework is mainly composed of 6 parts. (a) Contour Representation: Preprocessing and encoding of contour and category inputs. (b) Contour Fusion Module: Encourages more interaction between contour objects. (c) Global Information Injection: Adds global information representing contour diagrams to image features at different resolutions. (d) Contour Position Encoding: Treats image patches as square object contours, incorporating their position and shape information into a unified space, replacing traditional position encoding. (e) Object Space Awareness: Fusion of image and contour features without losing any token information, while utilizing self-attention layers to encourage interaction between different patches. (f) Objective function: The final objective function is constructed based on alignment loss.

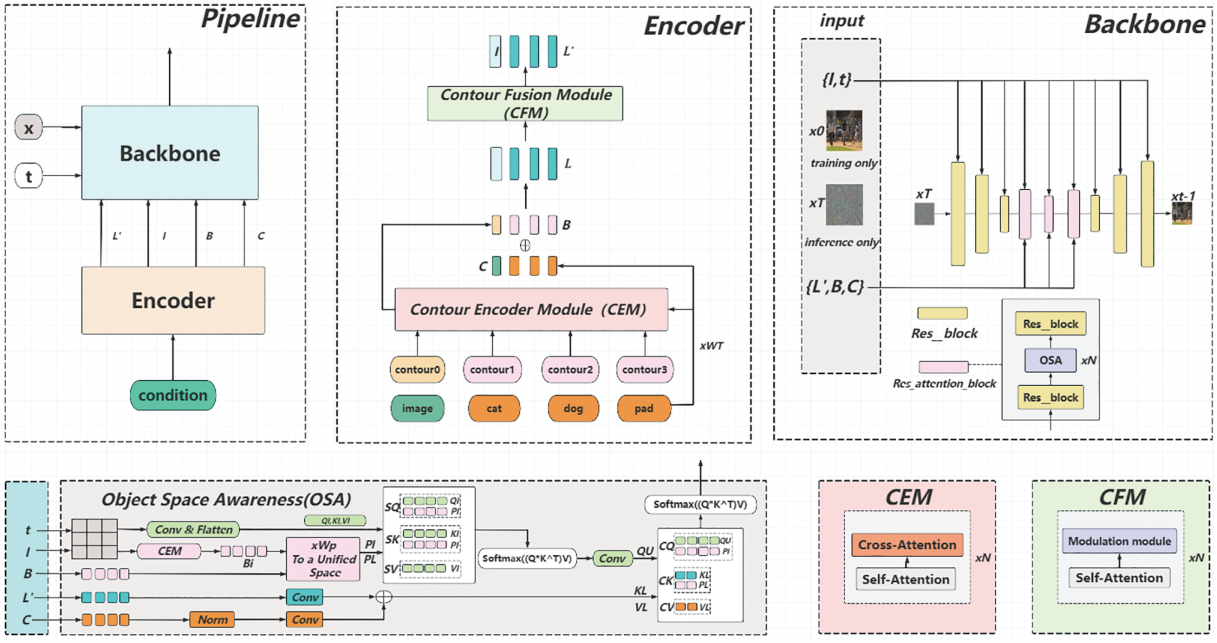

Figure 1: Illustration of the PolyDiffusion framework. The model jointly optimizes three objectives—denoising fidelity, contour alignment, and reward-driven refinement—to achieve precise control over multi-instance image generation

Each component plays a crucial role in achieving the goals of Poly Diffusion, from enhancing interaction between contours to incorporating global information and ensuring consistency in the diffusion process. The following subsections delve into the details of each component, elucidating their contributions and functionalities.

3.1 Contour Representation

We built this module inspired by Polyformer [21]. A contour, denoted as $C = \{O_1, O_2, O_3, \ldots, O_N\}$, is a collection of N objects, where each object is represented as $O_i = \{p_i, c_i\}$. The set $p_i = \{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}$ represents the polygon sequence of the object's contour using relative

coordinates. The set $P = \{p_1, p_2, \ldots, p_N\}$ is the collection of contour polygon sequences without the object categories c. Here, l denotes the number of contour points for each object.

Contours encode both object shape and position, offering richer structural detail than simple layout boxes. Unlike traditional bounding boxes or masks, which only provide coarse spatial information, contour representations capture the full geometric structure of each object. This allows PolyDiffusion to achieve precise control over the shape, size, and placement of multiple objects within a scene. To construct contour sequences b_i, we extract them from the dataset and interpolate them to a fixed length N, using placeholder representations $o_l = \{b_l, c_l\}$ for the global image and $o_p = \{b_p, c_p\}$ for objects without explicit contours. This process yields a unified set c containing N labeled contour instances. By decoupling object shapes from their spatial positions, contour representations enable better management of spatial relationships, thus overcoming the limitations of traditional methods that often struggle with shape distortion, spatial overlap, and semantic inconsistencies, particularly in complex multi-object scenes.

To mitigate starting-point ambiguity, each sequence is preprocessed to begin at the point nearest to the upper-left corner. Given the ring-like structure of contours, we introduce two encoding strategies: (1) a relative-position-based encoder incorporating self-attention for intra-contour relations and cross-attention for class-contour interaction, and (2) Rmamba, a cyclic scan module inspired by SimBA, designed to reduce the quadratic complexity of attention. We implement Rmamba by processing cyclically shifted sequences through shared-weight Mamba modules and aggregating the outputs, as shown in Fig. 2.

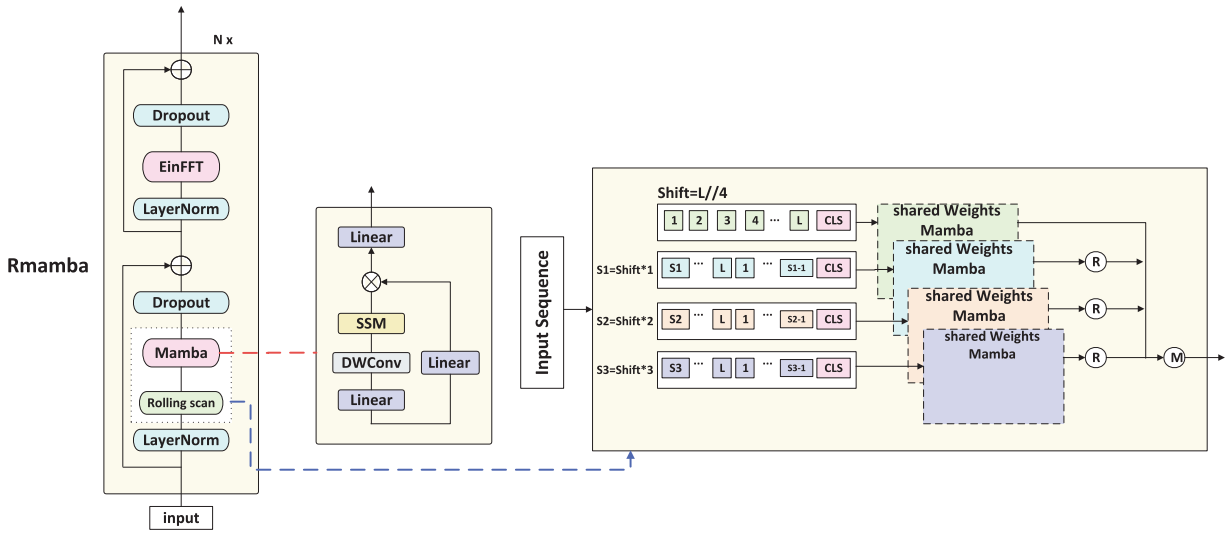

Figure 2: Rmamba consists of standard mamba blocks and a cyclic scan and EinFFT module

The contour encoding process treats each point (x_i, y_i) as a token, and prepends a center token $v = (x, y)$ representing the overall spatial context of the contour. This token is not part of the polygon itself but provides global shape information. The resulting sequence of $l + 1$ tokens consists of one global token and l contour point tokens. All points are projected into the embedding space via a shared linear layer. A ring-style relative positional embedding is added to the contour tokens to encode local structural order. The full sequence is then processed by the contour encoder.

For the class encoder, a simple linear layer $W_c \in \mathbb{R}^{1\times d}$ is implemented. By adding class features and contour features in the channel dimension, a comprehensive embedding L containing contour and class

information is obtained. The constructions are as follows:

$$L = B + C \tag{1}$$

$$C = cW_c \tag{2}$$

$$B = CEM((P \times W_B) + R, C) \tag{3}$$

To ensure effective structural encoding, we pre-train the Contour Encoder Module (CEM) on three contour-related tasks: (1) contour sequence reconstruction, (2) target frame prediction from contour input, and (3) object classification. We adopt a random masking mechanism that ignores specific coordinate tokens during attention calculation, enabling self-supervised learning without external labels.

Through these tasks, CEM learns not only the geometric structure of the contour but also contextual associations between the contour and the expected object appearance. This facilitates high-level semantic reasoning, which is critical for size-aware and location-aware image generation. Furthermore, the masking mechanism introduces controlled contour deformations, encouraging the model to develop robustness to incomplete or noisy contour inputs.

3.2 Contour Fusion Module

At this stage, each contour object is considered in isolation, lacking direct connections to other contour objects. To encourage interaction between different objects, we leverage the proximity of their centers within the image as an indicator of their relational closeness. Specifically, the closer the center points of two objects, the stronger the implied relationship between them. We aim to explicitly enhance this relationship.

Firstly, we calculate the original Attention Weight. Let the input token sequence be $L = (O_1, O_2, ..., O_n)$, and for each token O_j, we compute its original attention weight for each other token O_i as:

$$\mathbf{a_{ji}} = \frac{\text{Softmax}(Q(O_j) \times K(O_i)^T)}{\sqrt{d}} \tag{4}$$

We then adjust these weights based on a prior relationship graph. The distance between the center points of two objects is used to create an $n \times n$ relationship graph R, where R_{ij} represents the prior correlation between O_i and O_j. The adjusted attention weights are then given by $a_{ji} = a_{ji} \times (1 + \lambda \times R_{ij})$, with λ being a tunable parameter that governs the impact of prior knowledge. After this adjustment, the weights are re-normalized.

This method models the strength and weakness of relationships between objects using a heuristic approach. We believe that the approach is versatile and not limited to the distance between the center points of two objects. As an example, in scenarios with prominent occlusion, the approach relying on the IOU (Intersection over Union) value of object frames can be readily applied to gauge the intensity of occlusion between two objects.

The comprehensive embedding is then fed into the Contour Fusion Module, constructed from a multi-layer self-attention module. During the calculation of self-attention, the attention is adjusted according to prior knowledge. This module facilitates the fusion of contour features, incorporating relational information based on the proximity of object centers.

3.3 Global Information Injection

The Contour Fusion Module (CFM) is designed to aggregate contour-guided global and local representations into the image generation pathway. Specifically, we designate its first token O'_1 as a global contour

token that encodes the holistic structure of the input layout, while the remaining tokens correspond to individual object-level features.

To inject global structural priors into the image generation process, we propose a conditioning strategy that directly adds the projected global token to multi-resolution image features. Furthermore, to incorporate temporal awareness within the denoising process, we linearly combine the time step t with the global token, forming the final conditioning vector. This operation is formally defined as:

$$I' = I + O_1' W + t \tag{5}$$

where W is a learnable projection matrix, and I' denotes the image feature modulated by global contour context.

The resulting feature I' is embedded into the image generator using adaptive layer normalization [22], enabling hierarchical control at different levels of resolution. By injecting globally structured contour information, CFM enhances semantic consistency across instances and improves spatial alignment between contour guidance and visual synthesis. This module is crucial for enabling fine-grained instance control while maintaining coherence in layout-driven image generation.

3.4 Contour Position Encoding

The contour map inherently encapsulates spatial information, making it a pivotal component for integrating the spatial position, shape, and size of objects with image features. Rather than relying on generic positional embeddings, we propose a contour-aware strategy that encodes object-level structure directly from annotated boundaries.

Specifically, we treat each image patch as a square region associated with its corresponding contour coordinates. This allows contour tokens and patch tokens to share a unified spatial frame, enabling structural features to interact directly with visual features under the same coordinate system. This approach effectively replaces traditional position encoding with contour-based geometric guidance.

Let I represent the feature map of the image with height h, width w, and channel dimension d. For a patch located at the u-th row and v-th column, denoted as $I(u, v)$, we uniformly sample l points from its contour to define a boundary set:

$$S_{uv} = \{m_1, m_2, \ldots, m_l\}, \quad m_i = \left\{ \frac{x_i}{w}, \frac{y_i}{h} \right\} \tag{6}$$

This contour-based spatial encoding offers object-aware priors that reflect real-world structure, beyond the scope of simple coordinate indexing. By grounding positional semantics in actual boundary geometry, the model gains a more faithful understanding of spatial layout, which promotes better structure retention and spatial coherence in the generation process.

3.5 Object Space Awareness

Cross-attention and adaptive layer normalization have become standard in conditional generation. Following DIT [4], we adopt cross-attention due to its effectiveness in computing alignment loss. In text-to-image tasks, the first token often encodes global semantics but neglects individual token detail. This limitation is critical in contour-to-image generation, where each token carries distinct object-level information (shape, position, attributes); losing it risks pixel-level generation errors.

Object Space Awareness (OSA) is designed to aggregate two modes of information without losing any token information. By combining self-attention and cross-attention mechanisms, OSA enhances both

vertical and horizontal learning, ensuring that each object's spatial relationships and contextual information are preserved. To encourage interaction among different image patches, we first perform self-attention calculations within each patch, capturing the local structure and spatial coherence. This is followed by cross-attention calculations to allow the model to learn interactions between different image patches and objects, ensuring that spatial relationships between objects are maintained. By incorporating both self- and cross-attention, OSA ensures that each object's shape, position, and context are correctly aligned, thus improving spatial consistency and object interaction in complex multi-object scenes. This dual attention mechanism plays a crucial role in maintaining object-level semantics and structure fidelity during generation.

Specifically, we calculate S_Q, S_K, S_V on image patches to define O' as follows:

$$S_Q = \Phi(Q_I, P_I) \tag{7}$$

$$S_K = \Phi(K_I, P_I) \tag{8}$$

$$S_V = V_I \tag{9}$$

$$O' = \frac{\text{Softmax}(S_Q \cdot S_K^T)}{\sqrt{d}} \cdot S_V \tag{10}$$

Here, Q_I, K_I, and V_I represent Q, K, and V of image features, respectively, and Φ denotes concatenation on the channel dimension. Their structures are defined as:

$$Q_I, K_I, V_I = \text{Conv}(\text{Norm}(I)) \tag{11}$$

P_I and P_L represent the image patch contour embedding and object contour embedding, constructed as follows:

$$U = \text{CFM}(I) \tag{12}$$

$$P_I = U \cdot W_P \tag{13}$$

$$P_L = B \cdot W_P \tag{14}$$

where $W_P \in \mathbf{R}^{d \times d}$. Next, we calculate the cross-attention of image patches and contour embeddings, obtaining the definitions of C_Q, C_K, and C_V of O:

$$C_Q = \Phi(Q_U, P_I) \tag{15}$$

$$C_K = \Phi(K_L, P_I) \tag{16}$$

$$C_V = V_L \tag{17}$$

$$CA = \frac{\text{Softmax}(C_Q \cdot C_K^T)}{\sqrt{d}} \tag{18}$$

$$O = CA \cdot C_V \tag{19}$$

Finally, K_L and V_L represent K and V of global contour embeddings, respectively. Q_U represents the image patch after self-attention calculation, constructed as follows:

$$K_L = \text{Conv}(L') \tag{20}$$

$$V_L = \text{Conv}(\text{Norm}(C)) \tag{21}$$

$$Q_U = \text{Conv}(\text{Norm}(O')) \tag{22}$$

Here, CA is the cross-attention map, and all the CA of different sizes from various cross-attention layers, namely $8 \times 8, 16 \times 16, 32 \times 32$, are collected. The average cross-attention graph is calculated and normalized, followed by aggregation across multiple layers. Subsequently, it is converted to an approximately binary mask $\acute{CA}$ using a differentiable sigmoid function, and the DiceLoss with the real mask $mask_T$ is computed. This additional loss is then integrated into the final loss function.

The form of its loss function is as follows:

$$\mathcal{L}_2 = Dice(\acute{CA}, mask_T) \tag{23}$$

where $\acute{CA}$ is defined as follows:

$$\acute{CA} = LogisticFun(\check{CA}) \tag{24}$$

$$\check{CA} = \sum_{i=8}^{b} \alpha_i Up(CA_i) \tag{25}$$

where CA_i represents the cross attention graph with feature scale i.

OSA enhances the alignment between contour tokens and image patches by combining self-attention and cross-attention. This improves spatial precision during generation. The additional alignment loss further enforces contour-image consistency, making OSA essential for preserving object-level semantics and structure fidelity.

3.6 Objective Function

We employ a multi-objective optimization strategy, jointly optimizing three key losses via weighted summation: denoising diffusion loss for global fidelity, contour alignment loss for shape preservation, and reward-guided consistency loss (minimizing Fréchet distance) for semantic refinement. This balance ensures high-quality image generation with precise object-level control. PolyDiffusion thus effectively handles complex multi-object scenes without complex Pareto or game-theoretic modeling, offering a robust solution for multi-instance generation.

Diffusion Model: The diffusion model operates by progressively introducing noise to the image until the original signal is reduced to full Gaussian noise. Reversing this process allows converting random noise x_T into an image [23]. The diffusion process is governed by a Gaussian distribution and exhibits Markovian properties:

$$q(x_t|x_{t-1}) = \mathcal{N}\left(x_t; \sqrt{1-\alpha_t}x_{t-1}, \alpha_t I\right) \tag{26}$$

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\beta}}x_0, (1-\bar{\beta}_t)I\right) \tag{27}$$

Here, $\alpha_1, \ldots, \alpha_T$ form a fixed variance table, and we define $\beta_t := 1 - \alpha_t$ and $\bar{\beta}_t := \prod_{s=1}^{t} \beta_s$. The learning process involves optimizing a denoising device $x_t \sim q(x_t|x_0)$ to recover the original image. This is achieved by minimizing the loss function:

$$\mathcal{L}(\theta) = \mathbb{E}_{\epsilon,x,t}\left[\parallel \epsilon_\theta\ (x_t, t) - \epsilon \parallel^2\right] \tag{28}$$

In our approach, conditions are incorporated to generate corresponding objective functions:

$$\mathcal{L}_1 = \mathbb{E}_{\epsilon,x,c,t}\left[\parallel \epsilon_\theta\ (x_t, c, t) - \epsilon \parallel^2\right] \tag{29}$$

We introduce a single-step reward refinement inspired by ControlNet++. After freezing the base model post-initial training, we generate a clean sample x_0, simulate one forward diffusion step to obtain mildly noised $\tilde{x}_t \sim q(x_t \mid x_0)$, and optimize a reverse step to recover $\tilde{x}_0$. With small noise ϵ, this step approximates true denoising. We integrate this as an auxiliary loss in a narrow fine-tuning window.

In the fine-tuning phase, provided that the added noise ϵ is sufficiently small, it can be predicted that the original image $\acute{x_0}$ is equal to conducting one-step sampling on the perturbed image and merely performing the last step of denoising $\acute{x_t}$. Finally, we follow ControlNet++ 's bonus fine-tuning design:

$$\mathcal{L}_3 = \mathcal{L}(Img, \hat{Img}) = \mathcal{L}(Img, \theta(\acute{x_t}, c, t)) \tag{30}$$

Eventually, the loss consists of the diffusion training loss, the reward loss and the alignment loss:

$$\mathcal{L} = \begin{cases} \mathcal{L}_1 + \lambda_1 \mathcal{L}_2 + \lambda_2 \mathcal{L}_3, & \text{if } t \leq t_{\text{thre}}, \\ \mathcal{L}_1 + \lambda_1 \mathcal{L}_2, & \text{otherwise.} \end{cases} \tag{31}$$

In order to improve the quality of image generation, we use the classifier-free guidance technology. This method utilizes label condition guidance without a classifier, reducing training costs by combining conditional and unconditional networks in a single model. During training, the guide signal c is randomly discarded. After training, by adjusting the guidance intensity λ, the model can control the alignment degree of the guidance signal with the sample.

To achieve this, we construct an empty set contour c_ϕ containing the empty set contour point. During training, the model contour is diffused, and with a fixed probability, the condition for c is replaced by c_ϕ. During the sampling process, the image under contour conditions is sampled using the following formula:

$$\hat{\epsilon_\theta}(x_t, c, t) = \epsilon_\theta\ (x_t, c, t) + \gamma \cdot \epsilon_\theta\ (x_t, c_\phi, t) \tag{32}$$

We adopt the layout diffusion as the model backbone, a standard Unet network built using Resnet and transformer.

4 Experiments

We conduct our experiments based on the layout-to-image framework, LayoutDiffusion, using the following setup: the experiments are performed on a machine with 4 NVIDIA 3090 GPUs, running on Python 3.8 with PyTorch version 1.10.1, Torchvision 0.11.2, and Torchaudio 0.10.1. The CUDA Toolkit used is version 11.3, and the random seed for all experiments is set to 2024 to ensure reproducibility.

4.1 Datasets and Enhancement

We constructed our training and validation sets from the COCO-Stuff [24] and PASCAL VOC 2012 [25] benchmarks. From COCO-Stuff, we selected images exhibiting one to nine clearly delineated, non-facial object instances, yielding 28,292 training images and 1682 validation images; of these, 18,794 contained no human subjects. To maximize diversity while excluding portrait-centric scenes—given the specialized requirements and abundant data typically needed for high-fidelity face generation—we supplemented our non-human subset with images containing a single person alongside three to nine other objects. Our training protocol consisted of two phases: (1) an initial pass over the full COCO-Stuff dataset, and (2) a fine-tuning stage using only the "no-people" subset. We further applied standard data augmentation—random horizontal and vertical flips—to all images and their corresponding contour representations.

Notably, in order to strengthen the model's understanding of contours and boost its robustness in handling contour-related conditions, we specifically applied data augmentation to contour representation. After fixing the sequence of points to a predetermined length of l, we introduced random offsets to some points, allowing the contours to deform slightly.

4.2 Evaluation Metrics

We evaluate the performance of our model using several standard metrics. Fréchet Inception Distance (FID) measures the similarity between generated and real images by comparing their feature distributions using Inception V3. The Inception Score (IS) assesses the quality and diversity of generated images by evaluating the entropy of class and conditional distributions, also using Inception V3. Additionally, we use the Classification Accuracy Score (CAS) to evaluate the classifier's performance on the generated images by cropping and resizing ground truth bounding boxes and comparing them with real images through a trained ResNet-101 classifier.

To evaluate object detection performance, we calculate Average Precision (AP) and Average Recall (AR). These metrics compare instance masks from generated images with ground truth masks using YOLOv8m-Seg and COCO metrics. These evaluation techniques provide a comprehensive assessment of both image quality and instance-level generation precision.

4.3 Quantitative Results

As shown in Table 1, on COCO-Stuff at 128 × 128, PolyDiffusion reduces FID from 22.70 (PLGAN) to 16.71 and improves IS to 17.13 (vs. 12.20 for Grid2Im), a 40% gain. CAS reaches 45.22, up 3.7% over LayoutDiffusion, while AP_{mask} and AR_{mask} improve by 5.11 and 1.95 over InstanceDiffusion.

Table 1: For COCO-Stuff, we sample 8 images per condition and report quantitative results at 128 × 128

Methods	COCO-Stuff: 128 × 128				
	FID ↓	**IS↑**	**CAS↑**	AP_{mask} ↑	AR_{mask} ↑
Grid2Im [26]	59.50	12.20	4.05	–	–
PLGAN [27]	22.70	15.30	38.70	–	–
InstanceDiffusion [8]	–	–	–	48.62	36.71
LayoutDiffusion [17]	16.57	19.61	43.60	–	–
Ours	16.71	17.13	45.22	53.73	38.66

On VOC-2012 (128 × 128, Table 2), FID drops slightly to 15.94 and AP_{mask} increases to 54.72, outperforming LayoutDiffusion and InstanceDiffusion.

At 256 × 256 (Tables 3 and 4), COCO-Stuff FID improves to 15.56, IS to 25.14, CAS to 48.83, with AP_{mask}/AR_{mask} outperforming InstanceDiffusion by 11.5% and 3.0%. On VOC-2012, FID and AP_{mask} reach 17.56 and 52.17.

These results clearly demonstrate that, by *jointly optimizing* denoising fidelity, contour alignment, and reward-driven refinement, PolyDiffusion achieves balanced and significant gains in image quality, diversity, classification accuracy, and mask consistency—validating its superiority for complex multi-instance image generation.

Table 2: Quantitative results on VOC-2012 at 128 × 128 resolution

Methods	VOC-2012: 128 × 128	
	FID↓	AP_{mask} ↑
Grid2Im [26]	53.28	–
PLGAN [27]	19.14	15.30
InstanceDiffusion [8]	16.32	51.54
LayoutDiffusion [17]	16.17	19.61
Ours	15.94	54.72

Table 3: Quantitative results on COCO-Stuff at 256 × 256 resolution

Methods	COCO-Stuff: 256 × 256				
	FID ↓	IS↑	CAS↑	AP_{mask} ↑	AR_{mask} ↑
Grid2Im [26]	65.20	15.70	4.81	–	–
PLGAN [27]	29.10	18.60	37.65	–	–
InstanceDiffusion [8]	–	–	–	50.00	38.10
LayoutDiffusion [17]	15.61	27.61	47.74	–	–
Ours	15.56	25.14	48.83	55.73	39.25

Table 4: Quantitative results on VOC-2012 at 256 × 256 resolution

Methods	VOC-2012: 256 × 256	
	FID↓	AP_{mask} ↑
Grid2Im [26]	59.43	–
PLGAN [27]	21.33	–
InstanceDiffusion [8]	18.82	51.06
LayoutDiffusion [17]	17.94	–
Ours	17.56	52.17

These consistent gains across resolutions demonstrate that by jointly optimizing denoising fidelity, contour alignment, and reward-driven refinement, PolyDiffusion achieves balanced improvements in image quality, diversity, classification accuracy, and mask consistency.

5 Qualitative Results

Fig. 3 shows the qualitative results of PolyDiffusion, alongside a comparison with LayoutDiffusion. Our experiments demonstrate that PolyDiffusion outperforms LayoutDiffusion in both image variety and quality. While LayoutDiffusion generates diverse images, PolyDiffusion excels in maintaining high fidelity and consistency across instances.

On the COCO-Stuff dataset, PolyDiffusion reduces FID from 22.70 to 16.71 and increases IS to 17.13, achieving a 40% improvement over Grid2Im. On the VOC-2012 dataset, FID drops to 15.94 and APmask rises to 54.72, surpassing both LayoutDiffusion and InstanceDiffusion. At 256 × 256 resolution, COCO-Stuff's

FID improves to 15.56, IS increases to 25.14, and CAS reaches 48.83, with APmask improving by 11.5% over InstanceDiffusion. These results highlight PolyDiffusion's superiority in image quality and diversity.

Figure 3: Visualization of contour diffusion results. Each column shows the generated images under the same contour conditions, with comparisons between PolyDiffusion and LayoutDiffusion

The generated images in Fig. 3 further validate PolyDiffusion's ability to maintain high quality and consistency in multi-object generation, avoiding object overlap and distortion. In contrast, LayoutDiffusion struggles with shape distortion and spatial overlap in complex scenes.

Ablation Experiment

Tables 5 and 6 present ablation studies analyzing the contributions of individual components and their combinations. Table 5 compares different contour encoders and reveals that while attention-based encoders yield superior visual quality, Mamba-based methods—with linear complexity—offer a promising alternative for fine-grained contour representations due to their significantly lower resource demands [28]. Table 6 examines the effect of contour point granularity and shows that a finer-grained contour representation, achieved by increasing the number of contour points, improves the fidelity and accuracy of image generation, though at the cost of greater computational and memory requirements. These findings confirm that robust multi-instance image generation depends on the coordinated optimization of interaction priors, alignment constraints, and reward-guided refinement objectives.

Table 5: Ablation study on COCO-Stuff at 256 × 256

CFM	CEM	OSA	FID ↓	IS↑	CAS↑	AP_{mask} ↑	AR_{mask} ↑
×	✓	✓	18.63	22.92	42.19	53.41	32.54
✓	×	✓	17.47	19.14	40.85	51.92	31.36
✓	✓	×	28.65	15.62	36.23	47.13	26.45
✓	✓	✓	15.56	25.14	48.83	55.73	39.25

Table 6: Impact of different profile-description granularities on model performance

Points	64	96	128	Variable-length
FID↓	21.43	18.64	15.56	17.62
IS↑	19.36	23.21	25.14	23.71
AP_{mask} ↑	52.44	54.31	55.73	55.51
AR_{mask} ↑	38.13	38.54	39.25	38.62

6 Conclusion

We introduced PolyDiffusion, a contour-based diffusion framework that achieves precise instance-level control by jointly optimizing denoising fidelity, contour adherence, and reward-guided refinement. Extensive evaluations on COCO-Stuff and VOC-2012 confirm that PolyDiffusion outperforms current layout-to-image and mask-to-image methods in both image quality and instance precision.

Future work will focus on addressing PolyDiffusion's limitations in generating very small or highly fragmented objects, where low contour resolution and conflicts among objectives may impair fine structural reconstruction. To enhance fine-grained generation, we will explore adaptive contour sampling, multi-scale feature fusion, and instance-aware dynamic loss weighting. We also aim to support deployment in Convergence ICT scenarios through lightweight refinement modules and federated learning strategies that preserve user privacy and enable efficient edge–cloud collaboration.

Acknowledgement: I sincerely thank everyone who contributed to this paper; your dedication and insights were invaluable.

Funding Statement: This work is supported in part by the Scientific Research Fund of National Natural Science Foundation of China (Grant No. 62372168), the Hunan Provincial Natural Science Foundation of China (Grant No. 2023JJ30266), the Research Project on teaching reform in Hunan province (No. HNJG-2022-0791), the Hunan University of Science and Technology (No. 2022-44-8), and the National Social Science Funds of China (19BZX044).

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and system design: Yuzhen Liu, Jiasheng Yin; Supervision: Xiaoliang Wang; Data curation and preprocessing: Yixuan Chen, Jin Wang; Analysis and interpretation of results: Yuzhen Liu, Yixuan Chen, Jiasheng Yin; Draft manuscript preparation: Yuzhen Liu, Jiasheng Yin, Yixuan Chen, Jin Wang; Writing—review & editing: Xiaoliang Wang, Xiaolan Zhou. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: We used publicly available data and have referenced it in our paper; see references [24] and [25].

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Fan WC, Chen YC, Chen D, Cheng Y, Yuan L, Wang YCF. Frido: feature pyramid diffusion for complex scene image synthesis. In: Proceedings of the AAAI Conference on Artificial Intelligence. Washington, DC, USA: AAAI Press; 2023. Vol. 37, p. 579–87. doi:10.1609/aaai.v37i1.25133.
2. Creswell A, White T, Dumoulin V, Arulkumaran K, Sengupta B, Bharath AA. Generative adversarial networks: an overview. IEEE Sig Process Magaz. 2018;35(1):53–65. doi:10.1109/MSP.2017.2765202.

3. Esser P, Kulal S, Blattmann A, Entezari R, Müller J, Saini H, et al. Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first International Conference on Machine Learning; 2024 Jul 21–27; Vienna, Austria. p. 12606–33.
4. Peebles W, Xie S. Scalable diffusion models with transformers. In: Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision; 2023 Oct 1–6; Paris, France. p. 4195–205. doi:10.48550/arXiv.2212.09748.
5. Zhang J, Guo J, Sun S, Lou JG, Zhang D. Layoutdiffusion: improving graphic layout generation by discrete diffusion probabilistic models. In: Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision; 2023 Oct 1–6; Paris, France. p. 7226–36. doi:10.1109/ICCV51070.2023.00664.
6. Ding Y, Zhu X, Zou Y. Contour wavelet diffusion: a fast and high-quality image generation model. Comput Intell. 2024;40(2):e12644. doi:10.1111/coin.12644.
7. Qin C, Zhang S, Yu N, Feng Y, Yang X, Zhou Y, et al. Unicontrol: a unified diffusion model for controllable visual generation in the wild. arXiv:2305.11147. 2023. doi:10.48550/arXiv.2305.11147.
8. Wang X, Darrell T, Rambhatla SS, Girdhar R, Misra I. InstanceDiffusion: instance-level control for image generation. arXiv:2402.03290. 2024. doi:10.48550/arXiv.2402.03290.
9. Cao P, Yang L, Liu D, Yang X, Huang T, Song Q. What decreases editing capability? Domain-specific hybrid refinement for improved GAN inversion. In: Proceedings of the 2024 IEEE/CVF Winter Conference on Applications of Computer Vision; 2024 Jan 3–8; Waikoloa, HI, USA, p. 4240–9. doi:10.1109/WACV57701.2024.00417.
10. Cao P, Yang L, Liu D, Liu Z, Li S, Song Q. Lsap: rethinking inversion fidelity, perception and editability in gan latent space. arXiv:2209.12746. 2022. doi:10.48550/arXiv.2209.12746.
11. Kingma DP, Welling M. Auto-encoding variational bayes. arXiv:1312.6114. 2013. doi:10.48550/arXiv.1312.6114.
12. Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M. Hierarchical text-conditional image generation with clip latents. arXiv:2204.06125. 2022. doi:10.48550/arXiv.2204.06125.
13. Ashual O, Wolf L. Specifying object attributes and relations in interactive scene generation. In: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision; 2019 Oct 27–Nov 2; Seoul, Republic of Korea; 2019. p. 4561–9. doi:10.1109/ICCV.2019.00466.
14. Xie S, Zhang Z, Lin Z, Hinz T, Zhang K. Smartbrush: text and shape guided object inpainting with diffusion model. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023 Jun 17–24; Vancouver, BC, Canada, p. 22428–37. doi:10.1109/CVPR52729.2023.02148.
15. Couairon G, Careil M, Cord M, Lathuilière S, Verbeek J. Zero-shot spatial layout conditioning for text-to-image diffusion models. In: Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision; 2023 Oct 1–6; Paris, France, p. 2174–83. doi:10.1109/ICCV51070.2023.00207.
16. Nguyen Q, Vu T, Tran A, Nguyen K. Dataset diffusion: diffusion-based synthetic data generation for pixel-level semantic segmentation. In: Advances in neural information processing systems. La Jolla, CA, USA: NIPS; 2024. doi:10.48550/arXiv.2309.14303.
17. Zheng G, Zhou X, Li X, Qi Z, Shan Y, Li X. Layoutdiffusion controllable diffusion model for layout-to-image generation. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023 Jun 17–24; Vancouver, BC, Canada, p. 22490–9. doi:10.48550/arXiv.2303.17189.
18. Singh J, Zhang J, Liu Q, Smith C, Lin Z, Zheng L. Smartmask: context aware high-fidelity mask generation for fine-grained object insertion and layout control. arXiv:2312.05039. 2023. doi:10.48550/arXiv.2312.05039.
19. Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: transformers for image recognition at scale. arXiv:2010.11929. 2020. doi:10.48550/arXiv.2010.11929.
20. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, et al. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision; 2021 Oct 10–17; Montreal, QC, Canada. p. 9992–10002. doi:10.1109/ICCV48922.2021.00986.
21. Liu J, Ding H, Cai Z, Zhang Y, Satzoda RK, Mahadevan V, et al. Polyformer: referring image segmentation as sequential polygon generation. In: Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2023 Jun 17–24; Vancouver, BC, Canada. p. 18653–63. doi:10.1109/CVPR52729.2023.01789.

22. Huang X, Belongie S. Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the 2017 IEEE International Conference on Computer Vision; 2017 Oct 22–29; Venice, Italy. p. 1501–10. doi:10.48550/arXiv.1703.06868.
23. Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. Adv Neural Inform Process Syst. 2020;33:6840–51. doi:10.48550/arXiv.2006.11239.
24. Caesar H, Uijlings J, Ferrari V. Coco-stuff: thing and stuff classes in context. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 1209–18. doi:10.48550/arXiv.1612.03716.
25. Everingham M, Van Gool L, Williams C, Winn J, Zisserman A. The PASCAL visual object classes challenge 2012 (VOC2012) Results [Internet]. 2012 [cited 2025 Aug 6]. Available from: http://host.robots.ox.ac.uk/pascal/VOC/voc2012/.
26. Wang B, Wu T, Zhu M, Du P. Interactive image synthesis with panoptic layout generation. In: Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022 Jun 18–24; New Orleans, LA, USA. p. 7783–92. doi:10.1109/CVPR52688.2022.00764.
27. Abdelfattah R, Wang X, Wang S. PLGAN: generative adversarial networks for power-line segmentation in aerial images. IEEE Trans Image Process. 2023;32:6248–59. doi:10.1109/TIP.2023.3321465.
28. Gu A, Dao T. Mamba: linear-time sequence modeling with selective state spaces. arXiv:2312.00752. 2023. doi:10.48550/arXiv.2312.00752.