LaRender: Training-Free Occlusion Control in Image Generation via Latent Rendering

Xiaohang Zhan
Tencent
xiaohangzhan@outlook.com

Dingming Liu
Tencent
dingmingliu3722@gmail.com

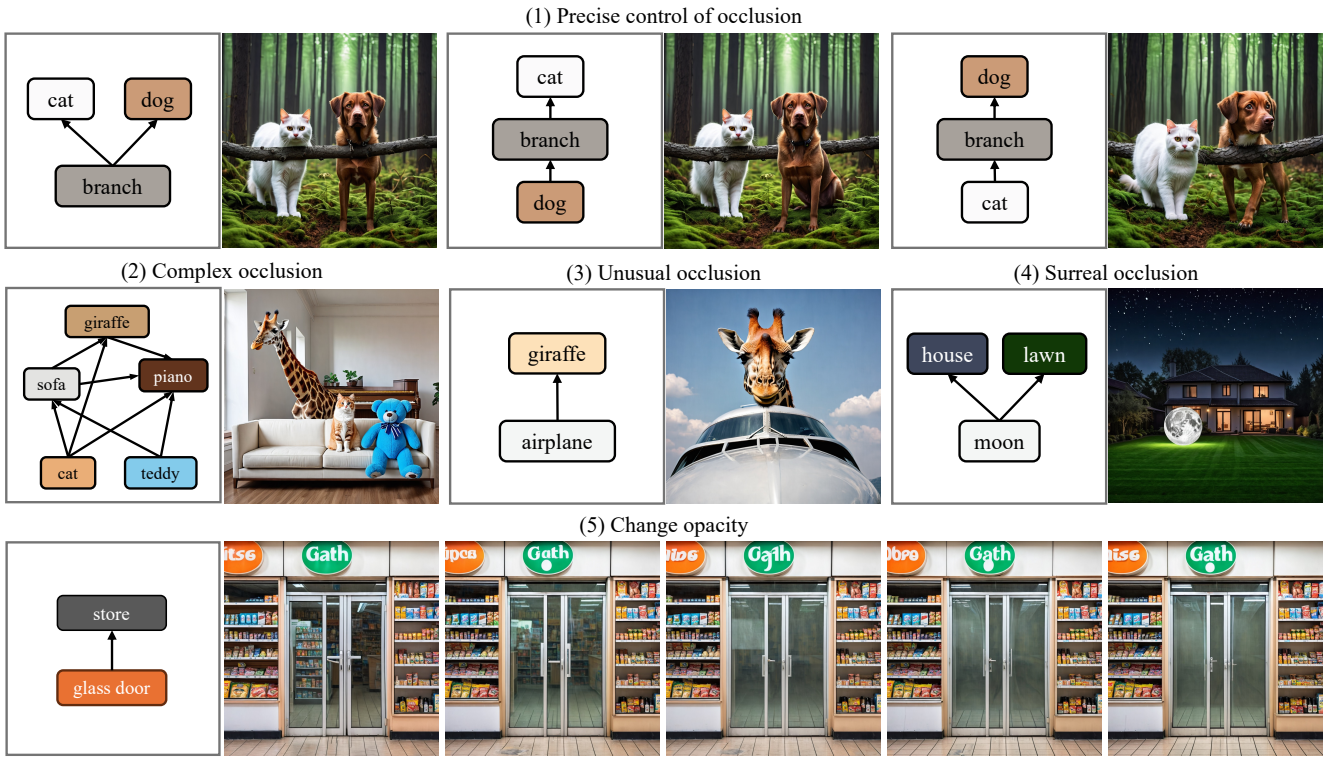

Figure 1. This work enables precise control of occlusion relationships among objects, and performs well in situations including complex, unusual and surreal occlusion. It also enables rich effects such as opacity control. All of these do not require training or fine-tuning. The background is omitted from the graph for clarity.

Abstract

We propose a novel training-free image generation algorithm that precisely controls the occlusion relationships between objects in an image. Existing image generation methods typically rely on prompts to influence occlusion, which often lack precision. While layout-to-image methods provide control over object locations, they fail to address occlusion relationships explicitly. Given a pre-trained image diffusion model, our method leverages volume rendering principles to "render" the scene in latent space, guided by occlusion relationships and the estimated transmittance of objects. This approach does not require retraining or fine-tuning the image diffusion model, yet it enables accurate occlusion control due to its physics-grounded foundation. In extensive experiments, our method significantly outperforms existing approaches in terms of occlusion accuracy. Furthermore, we demonstrate that by adjusting the opacities of objects or concepts during rendering, our method can achieve a variety of effects, such as altering the transparency of objects, the density of mass (e.g., forests), the concentration of particles (e.g., rain, fog), the intensity of light, and the strength of lens effects, etc.

1. Introduction

Occlusion control is a critical aspect of image generation, particularly in applications such as advertising content creation, concept design, and complex scene generation, where the spatial arrangement and interaction of objects must be accurately represented. Precise occlusion control ensures that objects are correctly layered and interact in a visually coherent manner, which is essential for creating realistic and immersive scenes.

Despite the advancements in image generation techniques, current state-of-the-art methods struggle to provide precise occlusion control. Existing text-to-image approaches [5, 22, 24, 25, 27–30] rely on text prompts to control occlusion, such as *"a dog behind a cat, and a bird is in front of the cat"*. However, in practice, surprisingly, even the state-of-the-art method [16] performs poorly in occlusion control, especially in complex scenes with multiple objects occluding each other.

The other related branch is layout-to-image generation [2, 6, 17, 36, 44–46]. Given the layout that is usually represented as bounding boxes, these methods can generate complex scenes. Since layout shares some common priors with occlusion, layout-to-image models have the potential to generate images with reasonable occlusion relationships. However, due to the complexity of the occlusion phenomenon, these methods cannot accurately control the occlusion relationships. As shown in Figure 1 (1), each object shares the same bounding box in different occlusion cases. Another trivial solution, though unexplored in existing work, could involve training or fine-tuning an image generation model conditioned on occlusion relationships. However, this approach requires additional paired data of images and ground-truth occlusions, which are often expensive to collect.

The occlusion phenomenon shares the same essence as 3D rendering, whether from the perspective of human vision or a camera. This insight leads us to explore the relationship between 3D rendering and occlusion control in image generation. By leveraging principles of Volumetric Rendering, we propose **LaRender**, a non-parametric training-free method that effectively addresses the occlusion control problem in image generation. As shown in Figure 1, LaRender achieves precise occlusion control without the need for retraining or fine-tuning the image generation model. The method performs well even in complex, unusual, or surreal occlusion scenarios. Inheriting its essence from rendering, LaRender also enables control over object opacity, producing rich effects such as changing the transparency of objects, the density of mass (*e.g.,* forests), the concentration of particles (*e.g.,* rain, fog), the intensity of light, and the strength of lens effects. Please find more results in Figure 5. These capabilities come as a"free lunch", requiring no additional data or training.

In brief, we propose a non-parametric Latent Rendering mechanism that replaces vanilla cross-attention layers of a pre-trained image diffusion model. Given an occlusion graph, our method first sorts objects from bottom to top and rearranges the objects' latent features, *i.e.,*, the hidden states from the denosing network, followed by a virtual camera facing the latent features. We then perform Latent Rendering, which borrows principles from volumetric rendering but operates at the latent level rather than the pixel level. In this way, latent features of objects are integrated according to physical rules that consider both occlusion relationships and object transmittance.

The key contributions of our work are as follows:

- Inspired by the shared essence of occlusion and 3D rendering, we propose a novel mechanism, Latent Rendering, which for the first time, addresses the occlusion control problem in image generation without the need for training.
- Beyond precise occlusion control, we observe high-quality, physics-grounded visual effects enabled by LaRender, which potentially inspire new applications.

2. Related Work

2.1. Image Generation

Text-to-Image Generation. Recent advances in diffusion models [5, 22, 24, 25, 27–30] have significantly improved text-to-image generation, enabling diverse, high-quality, and controllable outputs. Open-source projects like Stable Diffusion [29], Stable Diffusion XL [25], and FLUX [16] have further facilitated research in this area.

Layout-to-image Generation. Layout-to-image methods [2, 6, 17, 34, 36, 44–46] control object spatial layouts to generate complex scenes. Some [17, 45, 46] fine-tune diffusion models using datasets with bounding box annotations, such as MS COCO [18]. Specifically, 3DIS [46] generates depth maps first, then renders textures from depth. However, depth does not equate to occlusion [40], and object depths remain uncontrollable. Others [2, 6, 36] proposed training-free methods focusing on object positioning. While our method uses bounding boxes to compute transmittance, it prioritizes occlusion control over precise positioning.

Scene graph-to-image Generation. Scene graphs represent objects and relationships. There are also low-level relationships, such as "in front of, behind" that share some common concepts with occlusion. However, these concepts focus on relative spatial positioning rather than the occlusion from the photographic perspective. For example, "a boy standing in front of a girl" can be a scene where they stand face to face, one on the left and the other on the right, with no visual occlusion. Moreover, scene graph-to-image generation methods [14, 32, 37] rely on datasets that have scene graph annotations, limiting their applicability.

Multi-layer Image Generation. MULAN [33] introduces a multi-layer dataset with RGBA decomposed objects. Zhang *et al.* [41] fine-tune latent diffusion models on 1M transparent image layer pairs for transparent image generation. LayerFusion [7] generates foreground, background and blended images from text prompts. However, these methods are not designed for complex scenes with multiple occluding objects.

3D-Aware Image Generation. Previous works [3, 4, 9, 23, 31] integrate GANs with neural radiance fields for 3D-aware image generation. They transplant learnable volume or triplane representations in GANs, allowing the synthesis of multi-view-consistent images. However, they are typically limited to constrained domains such as faces, cars, or animals. Recently, ViewDiff [11] leverages multi-view supervision and volume rendering to generate 3D-consistent images in more diverse settings. In contrast, our approach introduces a non-parametric volumetric rendering mechanism applied in the latent space of diffusion models, enabling generating complex scenes in a training-free manner.

2.2. Occlusion Handling

Occlusion handling is a classic challenge in computer vision, particularly in complex scene understanding. Datasets with occlusion annotations [26, 48] and self-supervised methods [40] have laid the foundation of occlusion handling from the perspective of perception. Follow up works [15, 19, 20, 38, 39, 47] further improve de-occlusion. However, from the generative perspective, occlusion handling remains a challenge. We are the first to address this challenge in the generative domain.

3. Methodology

3.1. Preliminaries of Volume Rendering

Volume rendering [8] computes the accumulated color of a pixel via integral of the volume color density along the camera ray:

$$\mathbf{C} = \int_{t_n}^{t_f} T(t)\sigma(t)\mathbf{c}(t)dt, \text{where } T(t) = \exp(-\int_{t_n}^{t} \sigma(s)ds), \tag{1}$$

where t is the position along the camera ray with near bounds t_n and far bounds t_f, $T(t)$ denotes the accumulated transmittance along the ray from t_n to t, $sigma(t)$ is the volume density, $\mathbf{c}(t)$ is the color of position t.

NeRF [21] numerically estimates this continuous integral via quadrature, resulting in the formula as:

$$\hat{\mathbf{C}} = \sum_{i=1}^{N} T_i(1-\exp(-\sigma_i\delta_i))\mathbf{c}_i, \text{where } T_i = \exp(-\sum_{j=1}^{i-1}\sigma_j\delta_j), \tag{2}$$

where $i = 1, 2, ..., N$ is the sampled positions in $[t_n, t_f]$, $\delta_i = t_{i+1} - t_i$ is the distance between the adjacent samples.

3.2. Latent Rendering for Occlusion Control

The key idea of Latent Rendering is to adapt the volume rendering formula to "render" a stack of layered latent features of objects, resulting in a fusion of object latent features that physically ensures occlusion relationships.

Latent re-arrangement. As shown in Figure 2, given the occlusion graph and the bounding boxes of each object, we first apply topological sorting to sort objects from top to bottom, numbered as $1, 2, ..., N$. Then given a trained image diffusion model, for the l-th cross-attention layer, we modify it to allow the latent representation $\mathbf{R}^{(l)}$ to attend to the prompt of each object, resulting in N object-wise latent features $\mathbf{R}_i^{(l)}$. Next, we draw an analogy to Volume Rendering: we stack all $\mathbf{R}_i^{(l)}$ from back to front, and position a virtual orthographic camera above the top object, facing the latent features. Since the camera is orthographic, the distance between adjacent latent features is not necessarily defined - we omitted the distance δ_i in our rendering formula. We use $\mathbf{R}_i^{(l)}$ to replace the color of all grid sampling positions in Volume Rendering, and define semantic density σ_i (a scalar) to replace the volume density in Volume Rendering.

Transmittance map. While topological sorting reflects the occlusion order, the specific regions of occlusion are unknown. There regions are essentially determined by the transmittance map - the spatial distribution of transmittance for each layer of objects. To estimate the transmittance map, we primarily use bounding box masks provided by users or LLMs. However, transmittance can be inaccurate in the interval between the bounding box mask and the actual area occupied by the object. To address this, we reuse the cross-attention maps from the aforementioned cross-attention module. First, we extract the index of the subject token via Dependency Parsing [12], and normalize the cross-attention map of the subject token to rescale its minimum and maximum to 0 and 1. Then, we perform an element-wise multiplication of this normalized map with the bounding box mask, yielding the final transmittance map.

Latent Rendering. Afterwards, we formally define Latent Rendering, as follows:

$$\begin{aligned} \mathbf{R}^{(l+1)} &= \frac{1}{\mathbf{S}}\sum_{i=1}^{N}\mathbf{T}_i(1-\exp(-\sigma_i))\mathbf{M}_i\mathbf{R}_i^{(l)}, \\ \mathbf{S} &= \sum_{i=1}^{N}\mathbf{T}_i(1-\exp(-\sigma_i))\mathbf{M}_i, \\ \mathbf{T}_i &= \exp\left(-\sum_{j=1}^{i-1}\mathbf{M}_j\sigma_j\right), \end{aligned} \tag{3}$$

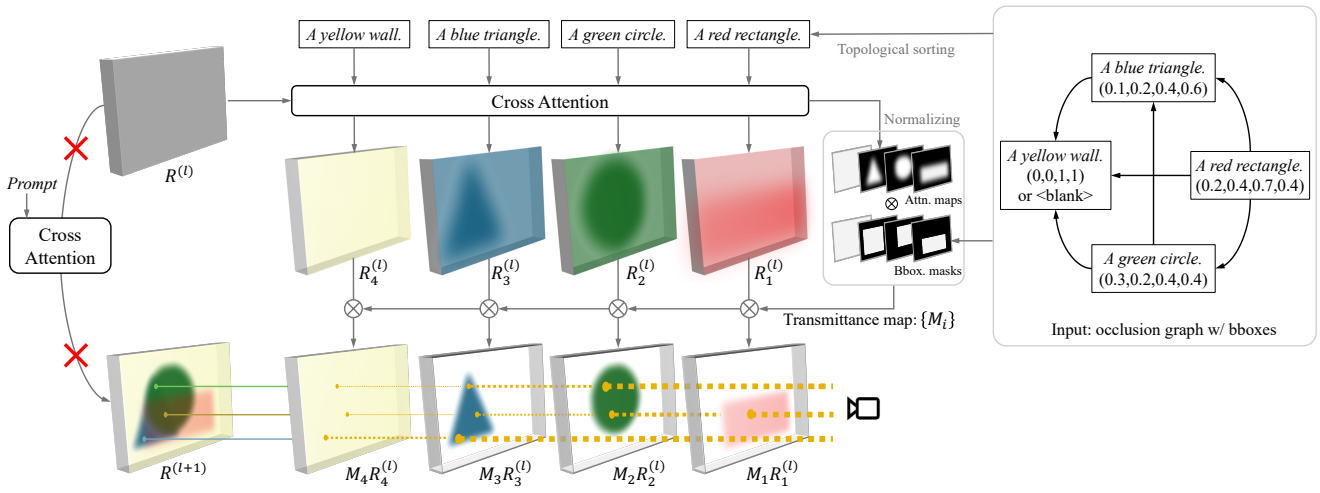

Figure 2. **LaRender** simply replaces vanilla cross attention layers to Latent Rendering layers. Given an occlusion graph with bounding boxes provided by users or parsed by LLMs, we rearrange these objects from back to front via topological sorting. For each cross-attention layer in a trained image diffusion model, we allow the input feature attend to each object prompt, obtaining object-wise latent features (hidden states). Then we estimate the transmittance map $\mathbf{M}_i$ for each object from its attention map and bounding box. Subsequently, we position an orthographic camera above the top object, facing the latent features. Finally, we apply the Latent Rendering formula to obtain the output scene representation that physically ensures occlusion relationships. The width of the orange dashed lines depicts the change of the accumulated transmittance $\mathbf{T}_i$ (initially to be 1) in different positions, and the colored solid lines represent the integrated latent features through their respective camera rays.

where $\mathbf{R}^{(l+1)}$ is the updated latent features to be fed into the next layer, $\mathbf{S}$ is the normalization term to prevent the output from deviating from the original latent distribution, $\mathbf{M}_i$ is the transmittance map, $\mathbf{T}_i$ is the accumulated transmittance map that describes the visibility of all pixels of object i from the virtual camera. Scalar $\sigma_i > 0$ is the semantic density of object i, similar to the volume density in volume rendering, we will discuss it later.

For methodological simplicity, we replace all cross-attention layers in the denoising network with our proposed Latent Rendering and follow the standard sampling steps to generate the image. Though we observed different behaviors of Latent Rendering in varying layers, we left it for future researchers to explore. Notably, Latent Rendering does not introduce any learnable parameters, making the framework entirely training-free. In this way, we generate images in a physically grounded manner that faithfully reflects occlusion relationships.

Density Scheduling. The semantic density σ_i, different from opacity, reflects the semantic strength of object concept i. We observed that in early denoising steps when the latent $\mathbf{R}_i^l$ does not yet have a strong concept of the object, Latent Rendering can disrupt this fragile concept by mixing the latent features. We also found that if $\sigma_i \to +\infty$, Latent Rendering degrades to "opaque mode" where all latent features become non-transparent, preventing them from mixing with each other. The mode is particularly suitable for early denoising steps. As revealed in [1], object concepts rapidly form during denosing, and late steps focus more on refining image quality rather than establishing concepts. Therefore, during middle and late steps, Latent Rendering can effectively distinguish different concepts even after mixing, and "opaque mode" is no longer required. Therefore, we introduce a fast-to-slow descent schedule to control the semantic density. Specifically, we define σ_i as a simple inverse proportional function of the diffusion step t:

$$\sigma_i(t) = \frac{D_i T}{T + 1 - t}, \tag{4}$$

where the scalar $D_i \geq 0$ is the semantic density of object i to be determined by users, and T is the total number of diffusion steps, $t = T, T-1, ..., 1$ is the current denoising step. During this schedule, initially $\sigma_i(T) = D_i T$ is a sufficiently large value, making Latent Rendering approximate an "opaque mode". At the last step, $\sigma_i(1) = D_i$, which means Latent Rendering converges to the target density D_i. We compare different schedules in the experiments.

Input mode. In cases where it is difficult for users to specify the occlusion graph and bounding boxes, we provide an alternative option: use an LLM to parse the original prompt to output the occlusion graph and the bounding boxes. We tried several LLMs, including those from GPT series, DeepSeek, and Claude, and observed over 95% accuracies in parsing the ordering, along with reasonable recommendation of bounding boxes. Thus, users can either simply write initial prompts or edit the occlusion and object

positions after the LLM's parsing. Besides, if the users do not wish to specify the background such as a *"wall"* in the case of Figure 2, they can use a blank prompt *""*.

3.3. Control Occlusion and Semantic Opacity

Via modifying the bounding boxes and the occlusion graph, we can easily control the precise occlusion relationships among objects, as well as rough positions. Note that though LaRender is able to control rough positions, it does not focus on improving the layout control accuracy - it is orthogonal to layout control methods.

Since the semantic density D_i is an unbounded positive value, we let $\alpha_i = 1 - \exp(-D_i)$ as the semantic opacity, and control $\alpha_i \in [0, 1)$ to observe the effects of semantic density. In extreme cases, when $\alpha_i = 0$, the object is entirely transparent and will disappear from the generated image; when $\alpha_i \to 1$, then $D_i \to +\infty$, that means the object is opaque with high density, and no pattern of other objects can appear through it.

When $\alpha_i \in (0, 1)$, does it mean the object is semi-transparent? Not exactly. Since we perform rendering in latent level rather than in RGB space, the semantic density should demonstrate high-level "semi-transparency". In fact, during our experiments, we observe interesting high-level "semi-transparency" phenomena, including changing density of mass (*e.g.*, forests), concentration of particles (*e.g.*, rain, fog), intensity of light, strength of lens effects, *etc*. Of course, if the semantic density of an object is strongly correlated with its transparency, then the physical transparency can naturally be controlled as well.

4. Experiments

4.1. Implementation Details

We used Stable Diffusion XL (SDXL) [25] pre-trained by IterComp [43] as the base model for its balance of quality and efficiency, though our method is compatible with varying diffusion-based image generation network. Results based on FLUX.1-dev [16] are included in the supplementary materials. To ensure fairness and avoid evaluation distractions, we did not employ prompt modification tricks or negative prompts. The only hyperparameter is the semantic opacity α for each object. We set $\alpha = 0.8$ for all quantitative comparisons and varied it from 0.1 to 0.9 in semantic density control experiments (Figure 5). All other inference hyperparameters, including the denoising step T in Equation 4, followed the base model's default settings. For occlusion graph and bounding box generation, we used DeepSeek R1 [10].

4.2. Evaluation Protocols

Baselines. There is no existing work that aims at occlusion control in image generation. However, text-to-image methods should be able to control occlusion via prompting, and layout-to-image models may generate images adhering to plausible occlusion relationships via carefully designed layouts, especially 3DIS [46] that uses depth as an intermediate signal. We maximized their possibility of occlusion control via prompt and layout design. The baselines we used are as follows:

1. SDXL [25]. A widely used text-to-image model.
2. FLUX.1-dev [16]. A state-of-the-art open-source text-to-image model.
3. MIGC [45]. A state-of-the-art layout-to-image model fine-tuned on the COCO dataset.
4. 3DIS [46]. A state-of-the-art layout-to-depth-to-image model fine-tuned on COCO. We used the FLUX version of the depth-to-image stage for the maximal quality.

Evaluation data and metrics. We used the data below:

1. **T2I-CompBench++** [13] - 3D spatial relationship evaluation. The validation split contains 300 prompts, each describing two objects with simple spatial relationships (*e.g.*, "in front of", "behind", "hidden by"). We used DeepSeek to generate bounding boxes and manually adjusted failure cases for methods requiring box inputs. We followed the standard UniDet-based metric, which combines object detection and depth estimation to infer object order.
2. **RealOcc.** To complement T2I-CompBench++, which lacks real layouts and only includes two-object prompts, we created RealOcc, a dataset with real-world bounding boxes. We curated images with 2-5 objects from the COCOA [48] validation set, filtering out extremely small or large bounding boxes and invalid annotations such as "background", ensuring each object has at least one occlusion relationship inferred from COCOA's occlusion annotations, and ensuring a balanced distribution of number of objects. This results in 70 examples with 249 objects with amodal bounding boxes and 261 occlusion pairs. While small, it serves as a valuable complement to T2I-CompBench++. Detailed statistics are provided in the supplementary materials.

Additionally, we conducted user studies as a more comprehensive evaluation of occlusion relationships. We designed 15 comparison samples per dataset and distributed over 30 questionnaires. Following ControlNet [42], we reported Average User Ranking (AUR) and Human-Perceived Success Rate (HPSR) in Table 1. To ensure Latent Rendering does not degrade the base model's generative ability, we also evaluated the CLIP score for both datasets. We did not compute FID, since the former dataset has no real images and the latter contains too few images that are unable to fulfill FID's minimal requirement of 3K images - it is hard to scale up this dataset due to the lack of amodal annotations.

Table 1. Quantitative comparisons on T2I-CompBench++ (3D Spatial) dataset and our proposed RealOcc dataset. Though there is no existing work on occlusion control, we maximized the ability of occlusion control of state-of-the-art text-to-image (SDXL, FLUX) and state-of-the-art layout-to-image (MIGC, 3DIS) models via prompt and layout design. Our method outperforms all baselines in occlusion related indicators,*i.e.*, "UniDet" and "User study", and shows minimal degradation in CLIP score. The denoising step is 25 for LaRender and SDXL, and 20 for FLUX and 3DIS. AUR and HPSR are calculated with 95% t-distribution Confidence Intervals.

Methods	Control	T2I-CompBench++ - 3D Spatial(val)				RealOcc			time (s)
		UniDet↑	User study		CLIP score↑	User study		CLIP score↑	
			AUR↑	HPSR↑		AUR↑	HPSR↑		
SDXL [25]	text	0.357	2.36 ± 0.49	0.322 ± 0.051	**31.12**	3.08 ± 0.40	0.396 ± 0.045	**29.42**	7.44
FLUX [16]	text	0.401	1.94 ± 0.42	0.293 ± 0.023	30.86	1.36 ± 0.25	0.244 ± 0.051	28.13	122
MIGC [45]	layout	0.373	3.56 ± 0.52	0.448 ± 0.068	30.73	3.31 ± 0.51	0.433 ± 0.070	28.49	11.1
3DIS [46]	layout	0.337	2.19 ± 0.33	0.311 ± 0.051	30.11	2.44 ± 0.49	0.344 ± 0.058	28.07	104
LaRender (ours)	occlusion	**0.416**	**4.94** ± 0.11	**0.767** ± 0.070	30.98	**4.81** ± 0.29	**0.767** ± 0.094	28.30	7.46

Table 2. The impact of different density schedules.

Schedule	Formula	UniDet↑	CLIP score↑
Fixed opaque mode	$\sigma_i(t) = D_i T$	0.240	28.32
Fixed density	$\sigma_i(t) = D_i$	0.393	30.44
Inverse proportional function (adopted)	$\sigma_i(t) = \frac{D_i T}{T+1-t}$	**0.416**	**30.98**

Table 3. The impact of the attention maps in computing transmittance maps.

Choice	UniDet↑	CLIP score↑
LaRender w/o attn. map	0.395	**31.00**
LaRender w/ attn. map	**0.416**	30.98

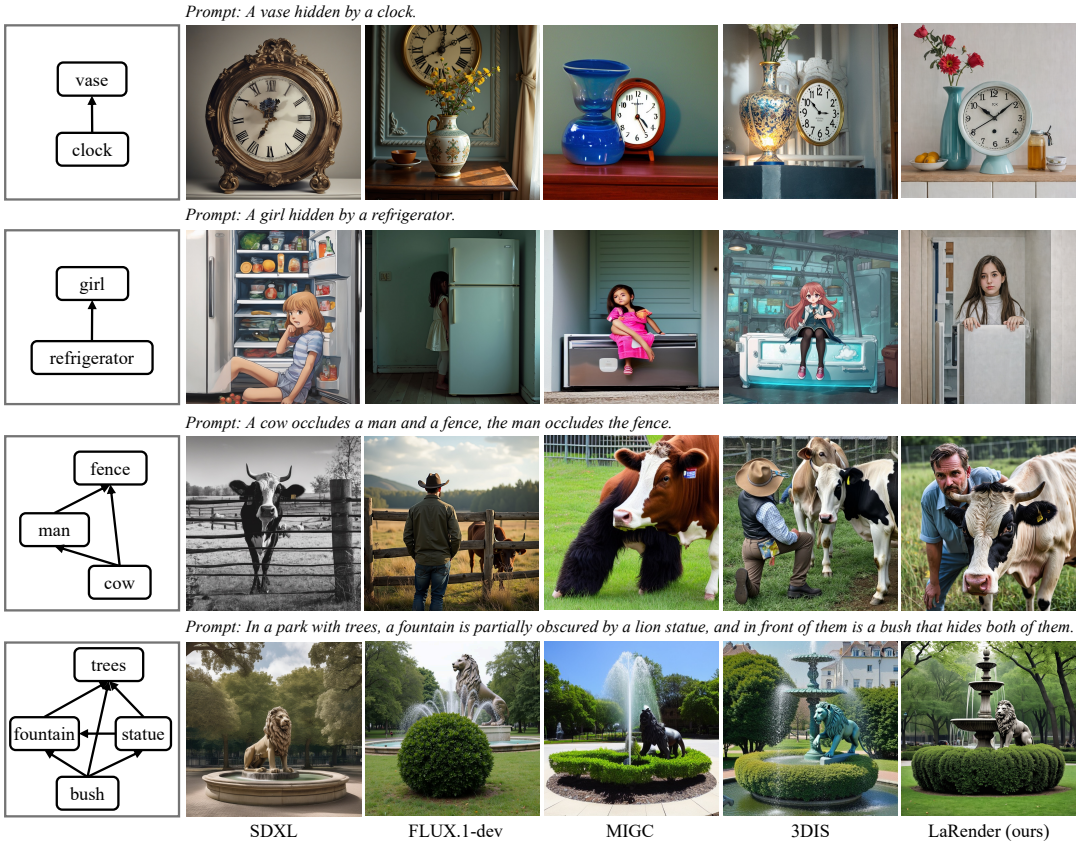

Figure 3. Qualitative comparison. The first column represents the occlusion graph. The cases in top two rows are results from T2I-CompBench++ (3D Spatial) dataset, the third row is a case in the RealOcc dataset, the case in last row is invented by ourselves.

Figure 4. Our generated results given similar layouts but different occlusion relationships.

4.3. Results and Comparisons

Quantitative comparison. As shown in Table 1, we obtained results of baseline methods on T2I-CompBench++ (3D spatial) and RealOcc datasets with their official code. Compared with state-of-the-art text-to-image and layout-to-image methods, LaRender achieves the best performance on occlusion related indicators, *i.e.*, UniDet and User study. Besides, we observe a slight decrease on CLIP score compared with SDXL. It is mainly because LaRender receives a list of objects as input, instead of a complete text prompt.

Qualitative comparison. As shown in Figure 3, for the cases with different numbers of objects, our method consistently produces precise occlusion controlled results. Though instructed by descriptions about occlusion, Text-to-image methods SDXL and FLUX cannot generate images with correct occlusion, especially when the number of objects is larger than two. Layout-to-image methods MIGC and 3DIS, though generate images with correct positions, cannot deal with occlusion relationships.

4.4. Analysis

Ablation study. We studied the effectiveness of density scheduling in Latent Rendering. As shown in Figure 2, in fixed opaque mode when $\sigma_i(t)$ is always a sufficiently large value, LaRender is degraded and the performance is poor. If we fix the density as a normal value during denoising, as discussed in the methodology section, the concepts can be destroyed, thus we observed decrease of the performance. Our adopted inverse proportional function performs the best on both UniDet and CLIP score. We also studied the impact of the cross-attention maps in computing the transmittance maps. As shown in Table 3, with attention map to shape the contour of objects in latent space, the transmittance map is more accurate, thus we observed higher performance in occlusion control, while bringing almost no impact on CLIP score. See supplementary materials for visualized results.

Time consumption. Compared with the base model SDXL, LaRender's increase of inference time is negligible. This is because the computation of LaRender involves element-wise multiplication and summation, which can be fully parallelized.

4.5. Applications

Controlling occlusion. With LaRender, we are able to freely control the occlusion relationships among multiple objects, even in similar layouts. As shown in Figure 4, given similar layouts but different occlusion relationships, LaRender consistently generates high-quality and occlusion-precise images. Please find more results including LaRender with FLUX in the supplementary materials.

Interactions. Normally LaRender accepts independent object prompts, we find that it supports interactions by assigning a full prompt to the background. The visual results are shown in the supplementary materials.

Adjusting Semantic Opacity. In addition to controlling occlusion relationships, LaRender exposes an adjustable parameter, the semantic opacity. By controlling the semantic opacities, we observed interesting "semi-transparency" phenomena. As shown in Figure 5, the semantic opacity affects the appearance of different objects in different ways, including changing the transparency of glass doors (Figure 1), the concentration of fog, the density of palm trees. Surprisingly, it can even be applied to non-object concepts,

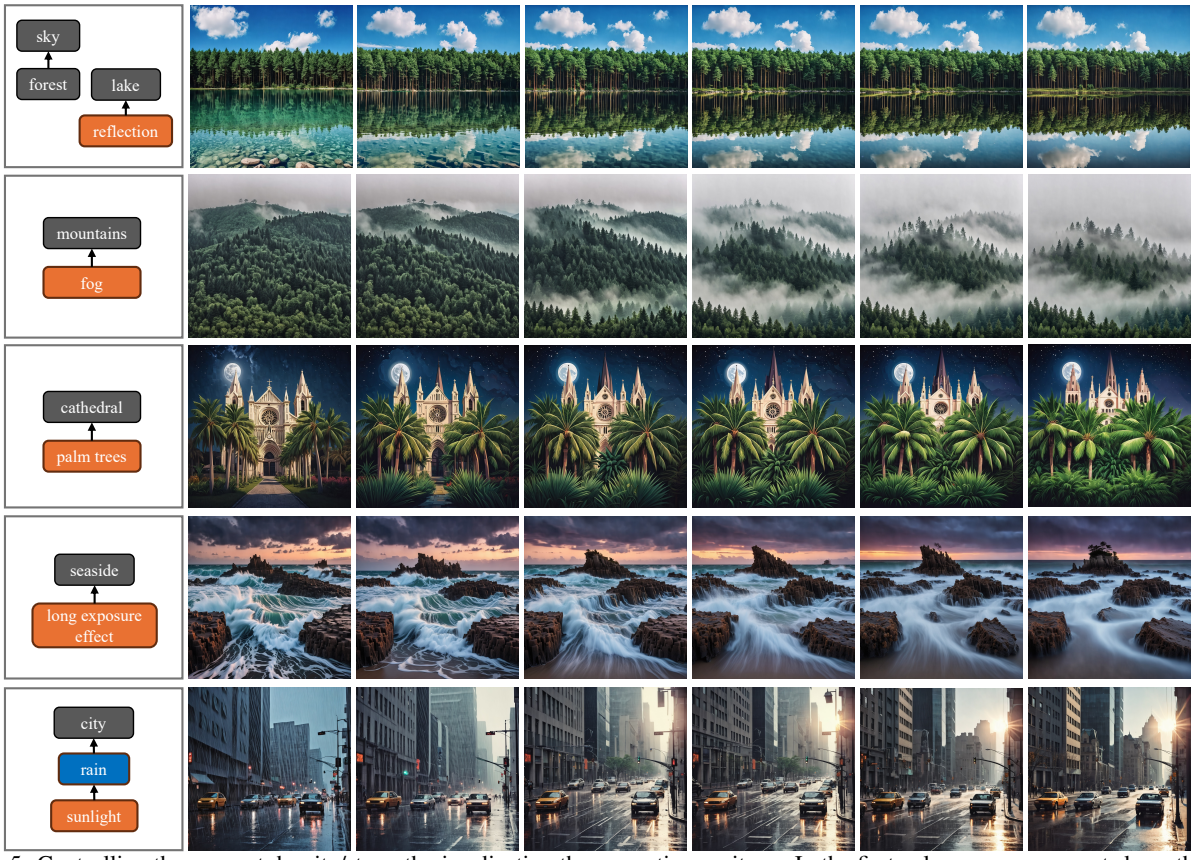

Figure 5. Controlling the concept density/strength via adjusting the semantic opacity α. In the first column, gray concepts keep the same opacities, orange concepts are increasing their opacities, and the blue concept is decreasing its opacity. Please find the dual-element grid figure for the last case in the supplementary materials.

Inaccurate position | Wrong occlusion | Concept lost | Concept mixed

Figure 6. Failure cases of LaRender. See Section 5 for analysis.

such as changing the strength of reflection, long-exposure effect and sunlight.

5. Conclusion, Limitations and Social Impact

In **conclusion**, we proposed a novel non-parametric mechanism, Latent Rendering, *abbr.* LaRender, that is able to provide precise control of occlusion relationships among objects in image generation in a training-free way. Extensive experiments prove its effectiveness and efficiency. We also observed interesting concept strength controlling effects brought by LaRender.

Limitations. The occlusion results can be wrong when the layout is not reasonable. The bounding boxes are rough hints of positioning, we are not pursuing accurate bounding box control in this paper. Additionally, sometimes the concepts can be lost or mixed, as shown in Figure 6. That is because the latent features can happen to be mixed or erased rather than partially occluded to satisfy its generative prior. Similar issues were discussed in [35].

Social Impact. The proposed method, as an image generation approach, could potentially be misused to create deceptive or misleading visual content, raising concerns about privacy, security, and misinformation. To mitigate these risks, we recommend implementing strict usage guidelines, monitoring for misuse, and developing robust detection mechanisms for identifying generated images.

References

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 4

[2] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. 2023. 2

[3] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5799–5809, 2021. 3

[4] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16123–16133, 2022. 3

[5] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 2

[6] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 5343–5353, 2024. 2

[7] Yusuf Dalva, Yijun Li, Qing Liu, Nanxuan Zhao, Jianming Zhang, Zhe Lin, and Pinar Yanardag. Layerfusion: Harmonized multi-layer text-to-image generation with generative priors. *arXiv preprint arXiv:2412.04460*, 2024. 3

[8] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. *ACM Siggraph Computer Graphics*, 22(4): 65–74, 1988. 3

[9] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021. 3

[10] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 5

[11] Lukas Höllein, Aljaž Božič, Norman Müller, David Novotny, Hung-Yu Tseng, Christian Richardt, Michael Zollhöfer, and Matthias Nießner. Viewdiff: 3d-consistent image generation with text-to-image models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5043–5052, 2024. 3

[12] Matthew Honnibal. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *(No Title)*, 2017. 3

[13] Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. 5

[14] 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. 2

[15] Lei Ke, Yu-Wing Tai, and Chi-Keung Tang. Deep occlusion-aware instance segmentation with overlapping bilayers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4019–4028, 2021. 3

[16] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 2, 5, 6

[17] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22511–22521, 2023. 2

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer, 2014. 2

[19] Huan Ling, David Acuna, Karsten Kreis, Seung Wook Kim, and Sanja Fidler. Variational amodal object completion. *Advances in Neural Information Processing Systems*, 33: 16246–16257, 2020. 3

[20] Zhengzhe Liu, Qing Liu, Chirui Chang, Jianming Zhang, Daniil Pakhomov, Haitian Zheng, Zhe Lin, Daniel Cohen-Or, and Chi-Wing Fu. Object-level scene deocclusion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 3

[21] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[22] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2

[23] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11453–11464, 2021. 3

[24] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 2

[25] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2, 5, 6

[26] Lu Qi, Li Jiang, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Amodal instance segmentation with kins dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2019. 3

[27] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021. 2

[28] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022.

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[30] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[31] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33:20154–20166, 2020. 3

[32] Subarna Tripathi, Anahita Bhiwandiwalla, Alexei Bastidas, and Hanlin Tang. Using scene graph context to improve image generation. *arXiv preprint arXiv:1901.03762*, 2019. 2

[33] Petru-Daniel Tudosiu, Yongxin Yang, Shifeng Zhang, Fei Chen, Steven McDonagh, Gerasimos Lampouras, Ignacio Iacobacci, and Sarah Parisot. Mulan: A multi layer annotated dataset for controllable text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22413–22422, 2024. 3

[34] Jiahao Wang, Caixia Yan, Weizhan Zhang, Haonan Lin, Mengmeng Wang, Guang Dai, Tieliang Gong, Hao Sun, and Jingdong Wang. Spotactor: Training-free layout-controlled consistent image generation. *arXiv preprint arXiv:2409.04801*, 2024. 2

[35] Tianyi Wei, Dongdong Chen, Yifan Zhou, and Xingang Pan. Enhancing mmdit-based text-to-image models for similar subject generation. *arXiv preprint arXiv:2411.18301*, 2024. 8

[36] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7452–7461, 2023. 2

[37] Ling Yang, Zhilin Huang, Yang Song, Shenda Hong, Guohao Li, Wentao Zhang, Bin Cui, Bernard Ghanem, and Ming-Hsuan Yang. Diffusion-based scene graph to image generation with masked contrastive pre-training. *arXiv preprint arXiv:2211.11138*, 2022. 2

[38] Guanqi Zhan, Weidi Xie, and Andrew Zisserman. A tri-layer plugin to improve occluded detection. *arXiv preprint arXiv:2210.10046*, 2022. 3

[39] Guanqi Zhan, Chuanxia Zheng, Weidi Xie, and Andrew Zisserman. Amodal ground truth and completion in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28003–28013, 2024. 3

[40] Xiaohang Zhan, Xingang Pan, Bo Dai, Ziwei Liu, Dahua Lin, and Chen Change Loy. Self-supervised scene de-occlusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3784–3792, 2020. 2, 3

[41] Lvmin Zhang and Maneesh Agrawala. Transparent image layer diffusion using latent transparency. *arXiv preprint arXiv:2402.17113*, 2024. 3

[42] 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. 5

[43] Xinchen Zhang, Ling Yang, Guohao Li, Yaqi Cai, Jiake Xie, Yong Tang, Yujiu Yang, Mengdi Wang, and Bin Cui. Itercomp: Iterative composition-aware feedback learning from model gallery for text-to-image generation. *arXiv preprint arXiv:2410.07171*, 2024. 5

[44] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22490–22499, 2023. 2

[45] Dewei Zhou, You Li, Fan Ma, Xiaoting Zhang, and Yi Yang. Migc: Multi-instance generation controller for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6818–6828, 2024. 2, 5, 6

[46] Dewei Zhou, Ji Xie, Zongxin Yang, and Yi Yang. 3dis: Depth-driven decoupled instance synthesis for text-to-image generation. *arXiv preprint arXiv:2410.12669*, 2024. 2, 5, 6

[47] Qiang Zhou, Shiyin Wang, Yitong Wang, Zilong Huang, and Xinggang Wang. Human de-occlusion: Invisible perception and recovery for humans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3691–3701, 2021. 3

[48] Yan Zhu, Yuandong Tian, Dimitris Metaxas, and Piotr Dollár. Semantic amodal segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1464–1472, 2017. 3, 5