EYE^3:Turn Anything into Naked-eye 3D

Yingde Song[1,*] Zongyuan Yang[1,*] Baolin Liu[1,*] Yongping Xiong[1,†] Sai Chen[1,*]
Lan Yi[1] Zhaohe Zhang[1] Xunbo Yu[1, ‡]
[1]Beijing University of Posts and Telecommunications, China
{songyingde,yangzongyuan0,baolin,ypxiong,chensai,ellinasen,chichi,yuxunbo}@bupt.edu.cn
* Equal contributions †First corresponding author ‡Second corresponding author

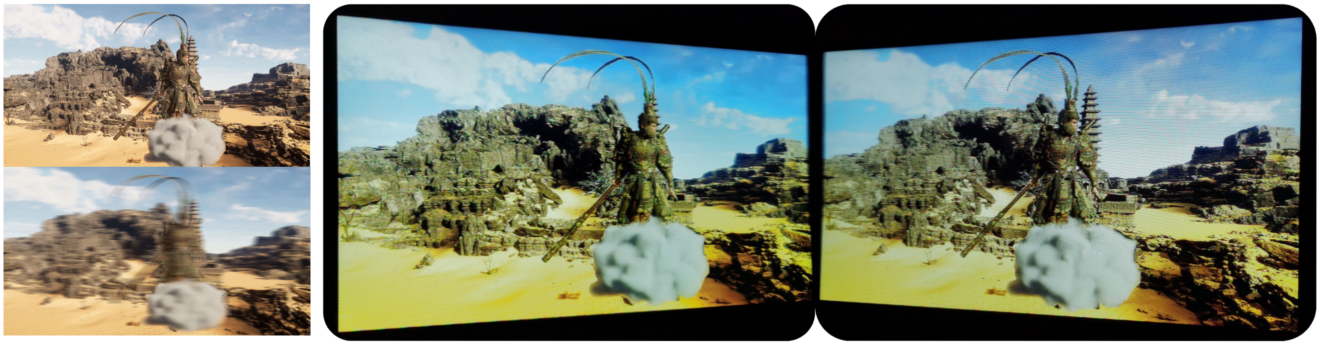

Figure 1. We present **EYE^3**, a framework that converts text, 2D images, and videos into high-quality light field display content with strong 3D consistency. This innovation delivers rich, realistic naked-eye 3D effects, addressing a major barrier to the widespread use of light field displays: the scarcity of 3D display content. The figure demonstrates the achieved effects: the left shows the original image and the autostereoscopic 3D content generated by **EYE^3**, while the right illustrates the actual effect on a light field display. Due to screen calibration limitations and capturing equipment, slight color shifts and moiré patterns may be observed.

Abstract

Light Field Displays (LFDs), despite significant advances in hardware technology supporting larger fields of view and multiple viewpoints, still face a critical challenge of limited content availability. Producing autostereoscopic 3D content on these displays requires refracting multi-perspective images into different spatial angles, with strict demands for spatial consistency across views, which is technically challenging for non-experts. Existing image/video generation models and radiance field-based methods cannot directly generate display content that meets the strict requirements of light field display hardware from a single 2D resource. We introduces the first generative framework **EYE^3** *specifically designed for 3D light field displays, capable of converting any 2D images, videos, or texts into high-quality display content tailored for these screens. The framework employs a point-based representation rendered through off-axis perspective, ensuring precise light refraction and alignment with the hardware's optical requirements. To maintain consistent 3D coherence across multiple viewpoints, we finetune a video diffusion model to fill occluded regions based on the rendered masks. Experimental results demonstrate that our approach outperforms state-of-the-art methods, significantly simplifying content creation for LFDs. With broad potential in industries such as entertainment, advertising, and immersive display technologies, our method offers a robust solution to content scarcity and greatly enhances the visual experience on LFDs.*

1. Introduction

Light Field Displays (LFDs) have experienced rapid advancements, offering a unique viewing experience distinct from that of Virtual Reality (VR) head-mounted devices[4, 5, 17, 20]. By utilizing a combination of LCD screens and cylindrical lens gratings, LFDs create a glasses-free 3D experience, directing different views to each eye and thereby producing a natural perception of depth [23, 26, 30]. The demand for 3D light field display (LFD) content is high, yet converting 2D media into compelling 3D representations presents significant challenges. Key to successful autostereoscopic 3D content creation are accurate depth representation and spatial consistency across multiple viewpoints. Despite LFDs' potential in sectors like entertainment, advertising, and immersive technology, the lack of high-quality autostereoscopic 3D content hampers their widespread adoption. This paper explores methods for generating high-quality autostereoscopic 3D content for LFDs from readily available 2D images, videos, or texts.

Previous research in video and image generation [21, 35] has made progress in synthetic content production. However, the unique requirements for generating content suitable for LFDs pose additional challenges. Each frame must be captured or rendered with precise camera poses to align with the grating parameters of the display [8, 19], necessitating a high degree of spatial consistency. Directly applying existing video and image generation techniques often results in artifacts and inconsistencies when viewed from varying angles, as these methods do not adequately consider the specific optical properties of LFDs. This realization underscores the need for a more tailored approach, which motivates our development of a generative framework designed specifically for LFDs.

In this paper, we introduce $\mathbf{EYE^3}$, the first generative framework explicitly crafted for 3D LFDs. Our method employs a point-based representation rendered through off-axis perspective techniques, ensuring precise light refraction and alignment with the optical requirements of the hardware. We fine-tune a video diffusion model to fill in occluded regions based on the rendered mask, thereby maintaining consistent 3D coherence across multiple viewpoints. This approach effectively overcomes the limitations of existing methods, which struggle to meet the stringent requirements of LFD hardware when starting from a single 2D resource.

We evaluated our method through experiments comparing $\mathbf{EYE^3}$ with state-of-the-art view synthesis techniques, using both quantitative metrics and user studies. The experiments involved synthesizing autostereoscopic 3D content from 2D images, videos, and texts, and assessing viewpoint control accuracy and completion quality against baseline methods. The results show that our approach outperforms existing methods, simplifying content creation for LFDs. In summary, our contributions are threefold:

1. We introduce $\mathbf{EYE^3}$, the first generative framework meticulously designed to meet the specific demands of 3D Light Field Displays (LFDs). This approach aims to contribute positively to the field of LFD content generation.
2. Our method combines a point-based representation with off-axis perspective rendering, enabling precise control over camera poses. Leveraging large video diffusion models, it enhances spatial and temporal consistency in multi-view synthesis, setting a new benchmark for converting 2D media into immersive 3D experiences tailored for LFDs.
3. Through extensive experiments and user studies, we validate the effectiveness of $\mathbf{EYE^3}$, demonstrating its superior performance and offering a promising solution to the challenges of LFD content creation.

2. Related Work

In this section, we first introduce the imaging principles of LFDs and highlight the challenge of content scarcity. We then review previous research aimed at enhancing LFD content, focusing on the conversion of 2D images into autostereoscopic 3D content as a solution to this issue. Next, we discuss advancements in novel view synthesis (NVS) with video diffusion models [29]. Finally, we explore the potential of using video diffusion models to generate autostereoscopic 3D content for LFDs.

2.1. 3D Light Field Displays

Since the emergence of LFDs, hardware advancements have significantly expanded their field of view and the number of viewpoints [38]. However, the issue of insufficient display content persists. LFDs work by refracting multi-perspective images, captured through cylindrical gratings, across various spatial angles, creating binocular parallax and a 3D effect. This process requires high spatial consistency among the multi-perspective images. Early efforts to supply content for LFDs focused on efficiently rendering mesh models, with techniques like Efficient Rendering [6] enabling real-time light field image processing on mid-range graphics cards. More recent advancements in 3D reconstruction, such as DirectL [34], have aimed to directly render NERF [18] and 3DGaussian [14] onto LFDs, increasing content diversity. Nevertheless, most digital assets remain in 2D formats. Our approach addresses this limitation by converting any 2D image, video, or text into high-quality LFD content, thereby alleviating the content scarcity issue.

2.2. Novel View Synthesis

With the development of diffusion models [11, 21, 25], these models have demonstrated remarkable ability in synthesizing high-quality images and generating new perspectives from a single image. Zero-1-to-3 [16] trains diffusion models on synthetic datasets with specific camera poses, enabling the synthesis of novel perspective images from a single view. However, this approach is limited to simple single-object images. ZeroNVS [22] builds upon this by incorporating real datasets, improving the model's generalization and performance in more diverse scenarios. Yet, it still encodes camera poses as text, limiting precise control over camera movement and leading to suboptimal consistency between synthesized images.

The integration of 3D modules into video diffusion models has improved view consistency but still lacks fine control over camera motion. MotionCtrl [28] attempts to con-

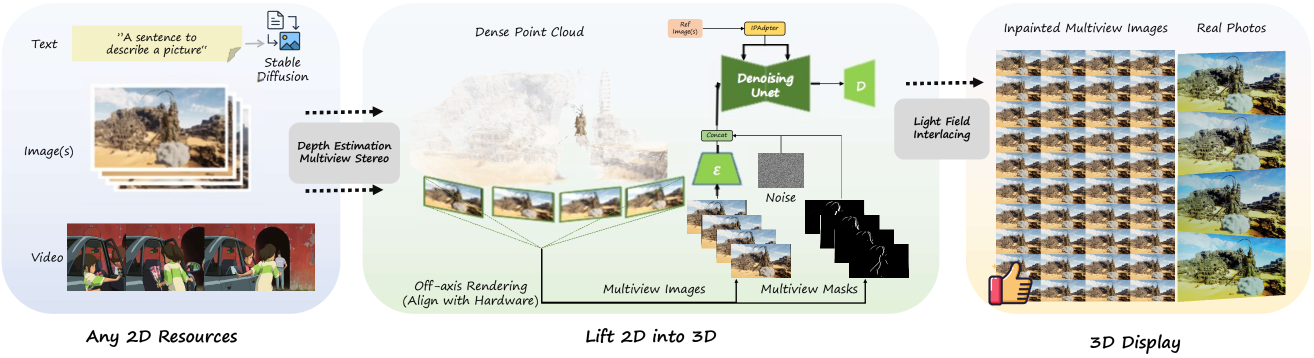

Figure 2. Overview of **EYE^3**. For any image, text-generated image, or video sequence frame, we perform depth estimation and use inverse perspective projection to restore the point cloud representation of the image. We then perform off-axis rendering for the view angles of LFDs to obtain multi-view images and their corresponding masks, and finally interlace according to the hardware parameters of LFDs to get autostereoscopic 3D content that can be displayed on LFDs.

trol camera movement through extrinsic parameters, but its reliance on 1D values does not allow for precise pose control. Methods like CamCo [31] and CameraCtrl [10], which incorporate Plücker coordinates [13], offer better control over camera movement but still struggle to handle extreme cases, such as narrow field-of-view angles or densely arranged cameras typical in LFDs. These approaches provide only basic control over the camera's position and orientation and cannot adjust internal parameters to achieve complex perspective effects, thus falling short of LFD-specific requirements for autostereoscopic 3D content [8, 19].

Inspired by 3D-Photo-Inpainting [24], which uses a mesh-based 3D representation to ensure multi-view consistency, we propose a point cloud-based 3D representation that allows for precise camera control, enhances perspective consistency, and satisfies the unique perspectival requirements of multi-view images on LFDs.

3. Method

3D-Photo-Inpainting [24] uses meshes as explicit 3D representations, but converting images into meshes often leads to adhesion between regions of different depths due to the continuous nature of mesh surfaces. While edge detection can help separate these regions, achieving complete and reliable separation remains challenging. In contrast, our method uses point clouds, whose inherent discreteness prevents such adhesion. This structure allows for more accurate and flexible rendering of masks for missing information, providing clearer guidance for inpainting and ensuring better consistency in the generated content. Furthermore, current commercial LFDS typically have a viewing field of less than 90 degrees, meaning they do not require the reconstruction of information from the rear or extreme angles. This limitation simplifies the inpainting task, as our approach focuses only on completing the visible portions of the scene, improving both efficiency and quality. By leveraging point clouds, we can effectively manage depth variations, ensuring high-quality, precise autostereoscopic 3D content tailored to these displays.

EYE^3 consists of three main steps (see Fig. 2): (1) First, monocular depth estimation and inverse perspective projection are used to convert the image into a point cloud for 3D representation (see Sec. 3.1). (2) Next, off-axis perspective rendering is applied to generate multi-view images and corresponding masks within a specified angular range (see Sec. 3.2). (3) Finally, a fine-tuned video model is employed to fill in gaps in the rendered multi-view images and masks (see Sec. 3.3). The completed multi-view images are then interlaced to produce the 3D display content required for LFDs, with the interlacing process following the method outlined in DirectL [34].

3.1. Image expansion and conversion to point cloud

For images, depth estimation is performed using Depth Anything V2 [33], and for videos, DepthCrafter [12] is used. During the conversion to point clouds, we assume the camera's principal point is at the image center. The exact focal length is not required; consistency with the focal length employed in Sec. 3.2 is sufficient. The spatial coordinates of each pixel are calculated using Eq. (1), (2), (3).

$$point_z = \frac{z \times Z_{max}}{\max(Z_{all})} + Z_{near}, \tag{1}$$

$$point_x = \frac{(W - 2 \times u) \times point_z}{2 \times f_x}, \tag{2}$$

$$point_y = \frac{(H - 2 \times v) \times point_z}{2 \times f_y}, \tag{3}$$

$point_x$, $point_y$, and $point_z$ respectively represent the x, y, and z coordinates of a point in space. The variable z represents the depth of the current point as obtained through

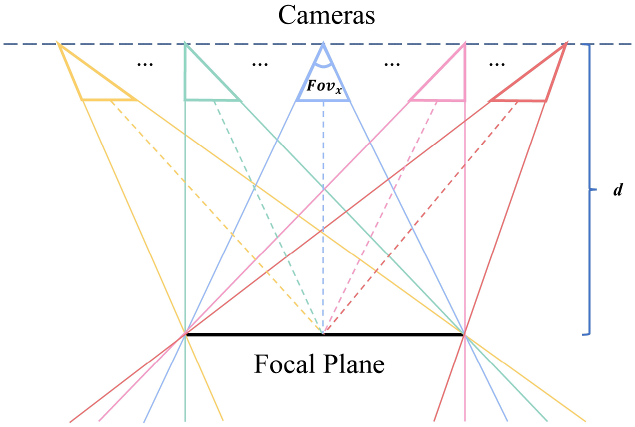

Figure 3. In off-axis perspective, cameras with the same focal length are positioned on the same horizontal plane, and each camera has a different optical center, allowing all cameras to simultaneously capture the same area of a specific horizontal plane in space. When displayed on a LFDs, areas of this plane that are farther from the cameras appear behind the screen space, while areas closer to the cameras are projected in front of the screen space.

depth estimation. Z_{all} denotes the collection of all points' depth values, and Z_{near} represents the depth of the camera's near plane. W and H respectively denote the width and height of the original image, while u and v represent the coordinates of the current point within the image. f_x and f_y represent the camera's focal lengths on the x and y axes, expressed in pixels. Z_{max} represents the maximum depth after normalizing the point cloud, which is related to the depth of entry and exit shown on the LFDs; the larger the value, the greater the depth of entry and exit, and vice versa [7]. The empirical formula we have derived from our experiments is: 6.2 × the metric depth of field can achieve relatively satisfactory performance on the screens used in the experiments.

By converting images into colored point clouds, we can obtain a preliminary explicit representation of the three-dimensional scene. Although relying on point clouds to capture multi-view images may face challenges such as voids and artifacts, this method can present complex perspective effects, fulfilling the special requirements of LFDs for off-axis perspective multi-view images. Moreover, because it provides highly consistent 3D information, it fundamentally ensures the 3D consistency of multi-view images.

3.2. Off-axis rendering based on point clouds

Due to the design of LFDs and the physiological structure of the human eye, the multi-view images required for autostereoscopic 3D content need to adopt an off-axis perspective. However, since the training data for video diffusion models rarely include off-axis perspective data, it is almost impossible to directly generate the required off-axis perspective content using large video diffusion models. Moreover, controlling the generation of dense viewpoints is also challenging. To address these, we use point clouds as the display representation and employ video models to fill in gaps after rendering the point clouds. During the rendering process, the cameras are positioned along the same line, and by adjusting the principal point parameters along the x-axis, all camera optical axes intersect at the same point as shown in Fig. 3. The camera's displacement at specified angles and the transformation of the x-direction principal points can be calculated using Eq. (4), (5).

$$C_x = (1+\frac{\tan(\theta)}{\tan(\frac{FOV_x}{2})}) \times \frac{W}{2}, \tag{4}$$

$$T = T + \begin{bmatrix} d \times \tan(\theta) \\ 0 \\ 0 \end{bmatrix}, \tag{5}$$

In this context, C_X represents the camera's principal point along the x-axis, expressed in pixels. θ represents the camera's rotation angle around the y-axis. W indicates the width of the imaging frame, d denotes the distance from the camera line to the center of the point cloud, T represents the camera's extrinsic matrix, and FOV_x denotes the camera's horizontal field of view. The relationship between FOV_x and f_x as mentioned in Sec. 3.1 is given by Eq. (6).

$$FOV_x = 2 \times \arctan(\frac{W}{2 \times f_x}). \tag{6}$$

3.3. Video inpainting based on large video models

Through the point cloud construction and rendering described in Sec. 3.1 and Sec. 3.2, we obtained off-axis perspective renderings of the point cloud and their corresponding masks. These renderings accurately represent the perspectival relationships between views with high 3D consistency. However, there are still issues with missing information in occluded regions. To improve the quality of multi-view images, our objective is to learn the conditional probability distribution $x \sim p(x|R, M)$ and generate high-quality multi-view images $x = \{x^0, \ldots, x^n\}$ based on the point cloud renderings R and their corresponding mask images M. Inspired by the exceptional quality and consistency of video diffusion models, we train a video diffusion model conditioned on the point cloud renderings and masks to approximate this distribution. The synthesis of new views can be interpreted as the inverse process, represented as $x \sim p_\theta(x|R, M)$, where θ represents the model parameters.

The point cloud image completion model based on video diffusion, as illustrated in Fig.2, builds upon the LDM architecture [21]. It incorporates a VAE encoder ϵ and decoder D for image compression, a denoising U-net network equipped with spatial and temporal layers, and an IP-

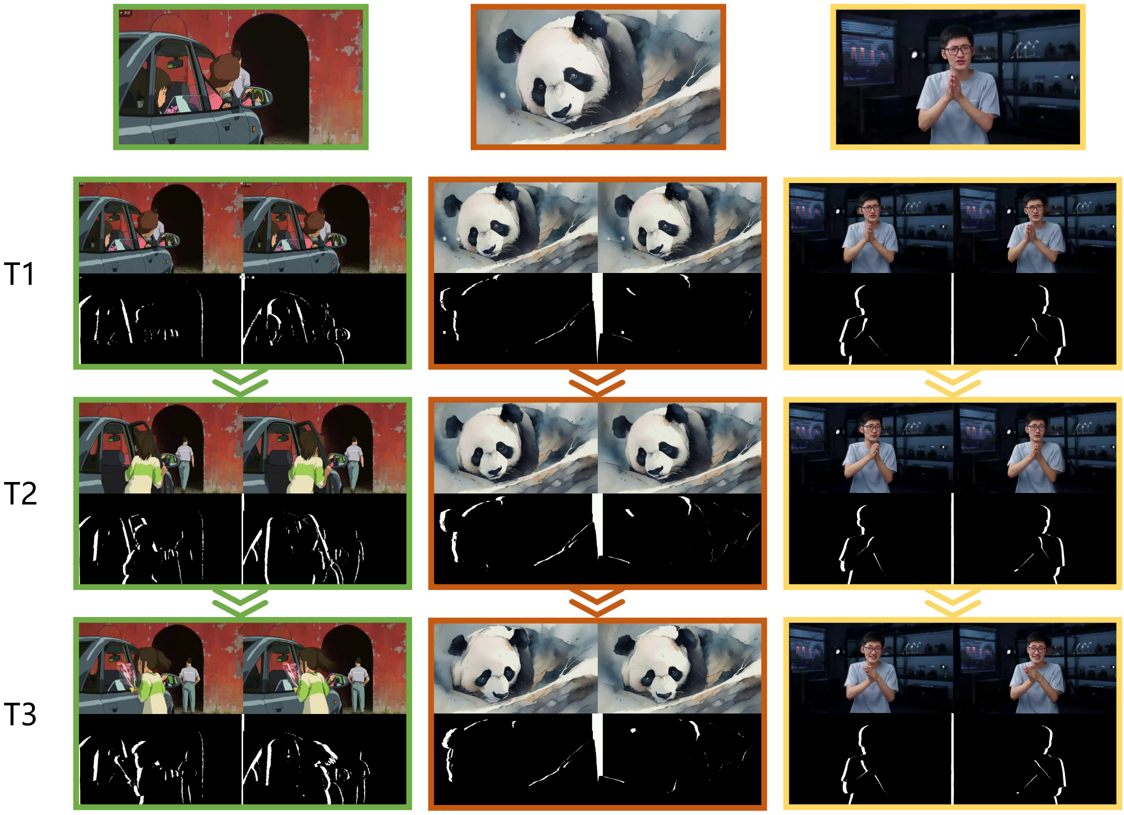

Figure 4. Three video clips from different sources are shown(from left to right is in the order of painting, AI, and photography.), with the first row being the first frame of the original video and the rows below representing three different moments. The left and right images are the new perspectives generated by our method for the three video clips, along with the corresponding masks produced by the point cloud rendering.

Adapter [36] for understanding reference images. After the rendered point cloud images are encoded by the VAE encoder ϵ, the encoded features are concatenated along the channel dimension with noise and the corresponding mask, and then passed into the denoising U-net network.

We fine-tune the open-source video model AnimateDiff V3 [9] by adding a mask dimension to the input of the denoising U-net network, enabling mask-guided control for completion. However, due to the scarcity and difficulty of creating point cloud-to-video datasets, alternative methods are required to generate datasets with similar data distributions for training.

We observed that missing regions in the multi-view images rendered from point clouds typically occur along the edges of the image content. To simulate this phenomenon, we first perform edge detection on videos and then apply dilation to the detected edges, thereby approximating the missing content distribution in multi-view images rendered from point clouds. These two distributions exhibit a high degree of similarity.

Using an 8-card A800 GPU setup, we fine-tuned the model on a dataset constructed with RealEstate10K [39]. Notably, compared to video inpainting models without mask guidance, such as Viewcrafter [37], our approach avoids the undesired redrawing of darker, non-missing regions, which often leads to low-quality results deviating from the original image.

In the process of converting videos to autostereoscopic 3D content, we handle each frame individually. Although there is no dedicated module to ensure the continuity of the inpainted content between frames, the explicit 3D representation conversion and rendering methods based on point clouds, as described in Section 3.1 and Section 3.2, ensure consistency across most areas. Furthermore, by introducing mask control during the completion process and considering the characteristics of the areas that need to be completed, we can still achieve stable and flicker-free autostereoscopic 3D content, as shown in Fig. 4.

4. Experiments

4.1. Experimental Setup and Implementation

We compared multiple state-of-the-art NVS methods with our approach. For SV3D [27], we utilized the `sv3d_p`

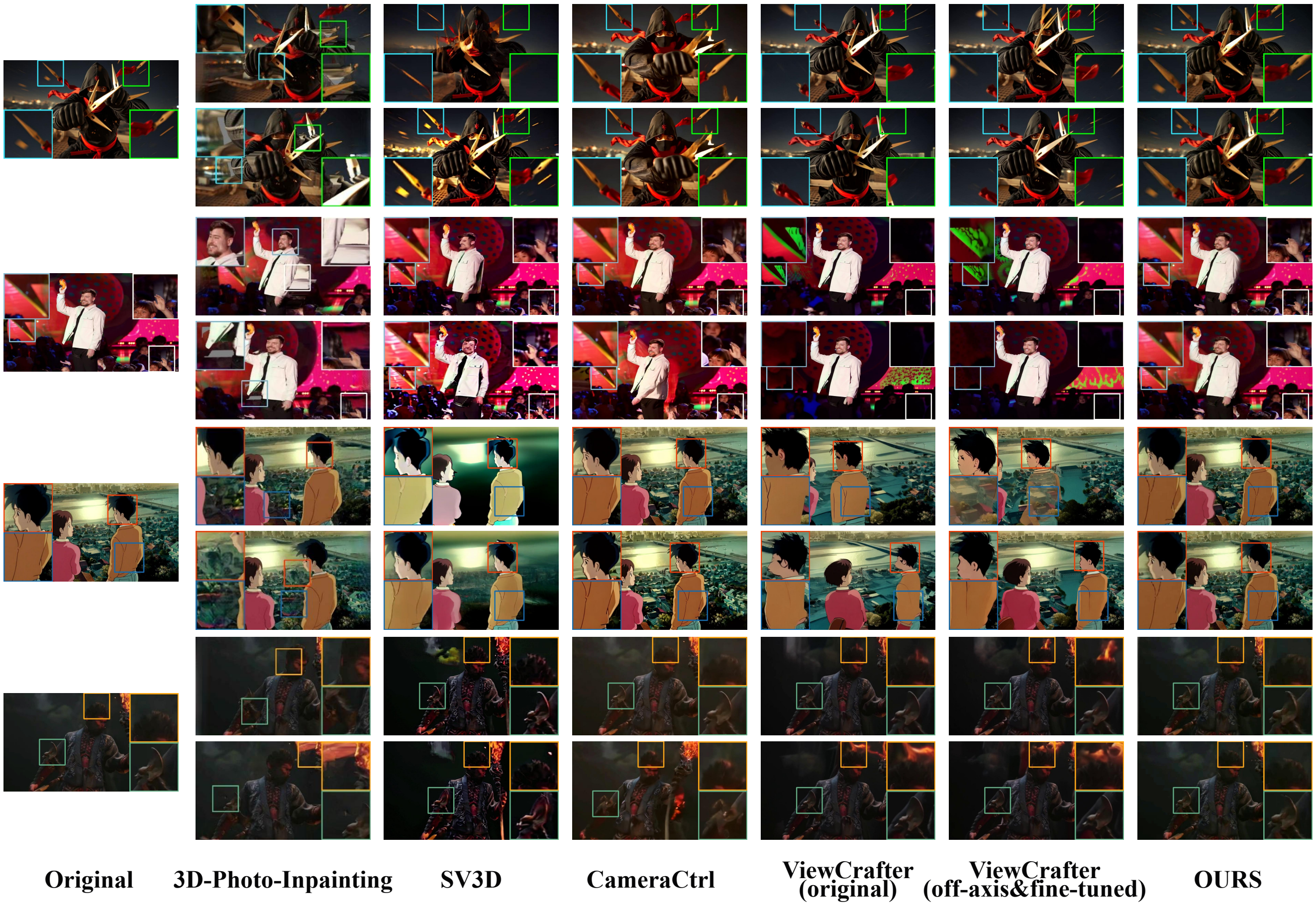

Figure 5. Qualitative Comparison of Multi-view Image Generation. The leftmost column shows the original reference images, while the two rows on the right sequentially display the leftmost and rightmost viewpoints of the generated multi-view images. From top to bottom, the four images represent different sources: AI-generated text-to-image, real-world photography, hand-drawn illustrations, and 3D modeling/rendering in games. To facilitate the observation of differences, we have marked and magnified certain details in the figures. Since the results of 3D-Photo-Inpainting have inconsistencies compared to other methods, the marked positions are also different.

weights for custom camera trajectory inference. For CameraCtrl [10], we employed the pretrained model on SVD [2]. For Viewcrafter [37] (original), we used the highest-quality weights at a resolution of 576x1024. Since these three methods do not support off-axis rendering, we employed a circular camera trajectory to uniformly capture 40 viewpoints within a range of -20° to 20°. We modified the official code of Viewcrafter [37] (off-axis & fine-tuned) and 3D-Photo-Inpainting [24] to implement off-axis perspective. In this process, 3D-Photo-Inpainting adopted Depth Anything V2 [33] for depth estimation to achieve more accurate depth results, and Viewcrafter(off-axis & fine-tuned) is fine-tuned on our dataset. Both Viewcrafter (off-axis & fine-tuned), 3D-Photo-Inpainting, along with our method captured 40 viewpoints within the same range of -20° to 20° using off-axis rendering.

Due to the challenges of collecting real-world off-axis perspective data, we constructed an evaluation dataset comprising 10 indoor and 10 outdoor scenes. Using Blender, we rendered these scenes into multi-view images to evaluate different methods using PSNR and SSIM metrics. However, on LFDs, achieving high 3D consistency across multi-view images is crucial, as any inconsistency significantly degrades the viewing experience. Consequently, while PSNR and SSIM provide some indication of the quality of the generated autostereoscopic 3D content, they cannot fully reflect the performance of the content on LFDs. Evaluating visual quality from a user perspective better aligns with the core principle of 3D display: if it looks correct, then it is correct.

To this end, we conducted a user study involving 30 unbiased participants. Although subjective to some extent, this approach more accurately reflects the actual display performance on LFDs. Participants evaluated each 3D image based on three criteria: (1) 3D Effect (the perception of objects moving in and out of the screen), (2) Comfort (the presence or severity of symptoms such as dizziness,

Table 1. PSNR and SSIM on Blender-Rendered Dataset, and Warp-e Metric for autostereoscopic 3D content Conversion from AI-Generated, Hand-Drawn, Photographic, and Game-Rendered Images.

	Indoor		Outdoor		warp-e			
	PSNR	SSIM	PSNR	SSIM	real	text	draw	game(3D render)
3D-Photo-Inpainting	15.3975	0.4102	12.6267	0.2581	2.38	2.25	2.57	1.62
SV3D	16.2388	0.5172	13.7280	0.3097	31.48	33.00	40.06	31.49
CameraCtrl	17.4618	0.5177	15.7407	0.4492	34.29	36.88	56.12	29.31
ViewCrafter (original)	19.5513	0.6188	15.8142	0.3969	76.14	22.59	30.69	31.57
ViewCrafter(off-axis & fine-tuned)	19.8743	0.6224	17.1147	0.4461	66.15	20.41	25.30	30.31
OURS	22.3988	0.7335	20.3076	0.5407	11.38	11.3	12.77	11.74

visual fatigue, or nausea), and (3) Image Quality (the similarity between the naked-eye 3D image on LFDs and the original 2D image, including consistency in key details and subjects). Each criterion was rated on a scale of 0 to 10. The final user-perceived quality score was calculated using a weighted formula based on these criteria (0.4, 0.3, 0.3).

Additionally, to quantify the consistency between multi-view images, we employed the warp error (warp-e) metric [15] using optical flow calculations.

4.2. Baseline Comparisons

Qualitative Analysis

The results are depicted in the Fig. 5, with the reference original image on the far left. 3D-Photo-Inpainting [24] uses meshes as explicit representations, but as mentioned in Sec.3,there are difficult-to-resolve mesh adhesion problems. Also, the completion effects on geometry and texture of mesh missing regions are low quality. SV3D [27], due to its training on the Objaverse dataset which primarily consists of data captured by circling around a single object, has limited capability for novel view generation in non-single object scenes and with small viewing angles. It is unable to maintain scene consistency and has incorrect camera poses. CameraCtrl [10], due to its control of the camera using only Plücker coordinates, simply maintains the relative poses between cameras and cannot ensure the absolute pose of the camera in the scene. In actual generation, the camera may appear reversed left-to-right, and the rotation angles may not reach the set angles, leading to imprecise camera control issues. ViewCrafter [37], whether using off-axis perspective and fine-tuned or not, due to the lack of mask control during image inpainting, it redraws the entire image, leading to spatial inconsistency in areas that already possess high spatial consistency, especially in darker regions, resulting in issues such as object movement and distortion. In contrast, our method can accurately control camera poses and implement complex perspective relationships while maintaining high spatial consistency in the completed image, avoiding issues like motion.

Quantitative Analysis

Firstly, we evaluate the accuracy of viewpoint control and the effectiveness of completion by calculating the PSNR and SSIM between the rendered results of the 3D mesh model and our outputs. Higher PSNR and SSIM values indicate that the viewpoints are closer to the camera views used for rendering in Blender and that the inpainted information aligns better with the rendered results. From Table 1, it is evident that our method significantly outperforms other approaches in both indoor and outdoor scenes.

For images obtained from photographs, hand-drawn sketches, or AI-generated graphics, where clear ground truth is unavailable, we use an optical flow-based warp error (warp-e) to evaluate the consistency between multiple views and measure the conversion quality. As shown in Table 1, our method is still better than most other methods except 3D-Photo-Inpainting. 3D-Photo-Inpainting [24] has high 3D consistency as it completes geometry and texture on the mesh and then renders it from different views. But 3D consistency is not the only factor of LDFs' display effectiveness. In our user study, Table 2, multi-view consistency and viewing comfort are directly related. But depth restoration and missing content filling are also key to LDFs' display.

Based on user feedback, in 3D-Photo-Inpainting, the multi-plane approach results in a weak 3D effect, also, low-quality mesh adhesion and filling make users feel the image quality is poor. SV3D [27] suffers from inaccurate camera pose control and poor 3D consistency, leading to weak 3D effects and discomfort during viewing. CameraCtrl [10] maintains relatively high 3D consistency but fails to achieve precise camera pose control, such as aligning to specific angles or following a designated sequence, which weakens the overall 3D effect. Viewcrafter [37], while capable of accurately controlling both off-axis and on-axis perspective camera poses, lacks explicit control over regional completion. Its reliance on global redrawing results in poor 3D consistency in the inpainted images, causing discomfort for viewers. In contrast, our method excels in both camera pose control and 3D consistency, achieving significantly higher user study scores than other methods.

Table 2. User Study Average Score, with the total score for each item indicated in parentheses.

	3D Effect	Comfort	Image Quality	Weighted Score
3D-Photo-Inpainting	4.767(143)	7.367(221)	2.833(85)	4.967
SV3D	3.533(106)	2.900(87)	1.833(55)	2.833
CameraCtrl	4.733(142)	5.667(170)	6.067(182)	5.413
ViewCrafter (original)	5.233(157)	5.167(155)	4.400(132)	4.963
ViewCrafter (off-axis & fine-tuned)	5.567(167)	5.667(170)	4.833(145)	5.377
OURS	6.833(205)	8.000(240)	8.167(245)	7.583

4.3. Ablation Studies

Exploring the Impact of Multi-View Image Inpaiting on Autostereoscopic 3D Content Quality. Our pipeline employs a fine-tuned video diffusion model to perform multi-view image completion. To evaluate the improvement in point cloud rendering image completion brought by mask-guided fine-tuning of the video diffusion model, we calculated PSNR and SSIM for indoor and outdoor scenes under three conditions: no inpainting, inpainting using a model without mask guidance, and inpainting using a fine-tuned model with mask guidance. The results are shown in Table 3.

The analysis reveals that models without mask guidance struggle to accurately preserve content outside the masked regions during inpainting. While the point cloud gaps are filled, the PSNR and SSIM metrics are even lower than those of the no inpainting condition. In contrast, the fine-tuned model with mask guidance not only effectively retains content outside the masked regions but also performs reasonable inpainting within the masked areas, achieving significantly higher scores.

Exploring the Impact of Depth Estimation Methods on Autostereoscopic 3D Content Quality. In our approach, any monocular depth estimation model can be used to estimate the depth of input images. We replaced the original monocular depth estimation module with three alternatives: DepthAnything [32], MiDaS V3.1 [1], and Depth Pro [3], and calculated the PSNR and SSIM for indoor and outdoor scenes. The results are presented in Table 4.

In indoor scenes, where depth variations are relatively small, all methods achieve comparable accuracy in depth estimation, resulting in similar scores. However, in outdoor environments characterized by large depth variations and open spaces, more precise depth estimation significantly improves the results, enhancing the quality of converting single images into autostereoscopic 3D content on the LFDs.

Table 3. Using PSNR and SSIM to Evaluate the Impact of Multi-View Image Inpainting on Autostereoscopic 3D Content Quality in Our Blender-Rendered Dataset

	Indoor		Outdoor	
	PSNR	SSIM	PSNR	SSIM
base(no inpainting)	22.1539	0.6979	18.7026	0.4892
base + no mask guid inpainting	20.7986	0.6842	17.8287	0.4305
base + finetuned mask guid inpainting(OURS)	22.3988	0.7335	20.3076	0.5407

Table 4. Using PSNR and SSIM to Evaluate the Impact of Depth Estimation Methods on Autostereoscopic 3D Content Quality in Our Blender-Rendered Dataset

	Indoor		Outdoor	
	PSNR	SSIM	PSNR	SSIM
base + MiDaS V3.1	22.2424	0.7013	18.7840	0.4887
base + Depth Pro	22.2008	0.7162	18.5279	0.5034
base + DepthAnything	22.3497	0.6797	19.2035	0.5371
base + DepthAnythingV2(OURS)	22.3988	0.7335	20.3076	0.5407

5. Conclusion

We propose a novel method that converts a single image into a 3D representation for display using monocular depth estimation. This approach enables precise control of camera poses and achieves complex perspective effects. Additionally, we leverage a fine-tuned video model to fill in missing regions of the image, generating dense multi-view images for synthesizing encoded patterns to be displayed on the LFDs. Our method addresses the limitations of existing approaches and provides a feasible solution to the scarcity of 3D resources for LFDs, contributing to its development and broader adoption.

5.1. Limitations and Future works

While our method ensures accurate 3D consistency and achieves complex perspective relationships through point cloud-based representation, meeting the content requirements of LFDs, it still has certain limitations. Its effectiveness is highly dependent on the accuracy of depth estimation, and in cases of inaccurate depth estimation, the results may fall short of expectations. Additionally, the use of a video model to inpaint multi-view images rendered from point clouds requires multiple iterations of denoising, which demands significant computational resources and results in longer inference times.

References

[1] Reiner Birkl, Diana Wofk, and Matthias Müller. Midas v3. 1–a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023. 8

[2] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 6

[3] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R Richter, and Vladlen Koltun. Depth pro: Sharp monocular metric depth in less than a second. *arXiv preprint arXiv:2410.02073*, 2024. 8

[4] Neil A Dodgson. Autostereoscopic 3d displays. *Computer*, 38(8):31–36, 2005. 1

[5] Netalee Efrat, Piotr Didyk, Mike Foshey, Wojciech Matusik, and Anat Levin. Cinema 3d: large scale automultiscopic display. *ACM Transactions on Graphics (TOG)*, 35(4):1–12, 2016. 1

[6] Laura Fink, Svenja Strobel, Linus Franke, and Marc Stamminger. Efficient rendering for light field displays using tailored projective mappings. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 6(1):1–17, 2023. 2

[7] Bangshao Fu, Xunbo Yu, Xin Gao, Xinhui Xie, Xiangyu Pie, Haoxiang Dong, Sheng Shen, Xinzhu Sang, and Binbin Yan. Analysis of the relationship between display depth and 3d image definition in light-field display from visual perspective. *Displays*, 80:102514, 2023. 4

[8] Ajinkya Sudhir Gavane. *Novel Applications of Multi-View Point Rendering*. North Carolina State University, 2023. 2, 3

[9] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 5

[10] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024. 3, 6, 7

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[12] Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. Depthcrafter: Generating consistent long depth sequences for open-world videos. *arXiv preprint arXiv:2409.02095*, 2024. 3

[13] Yan-Bin Jia. Plücker coordinates for lines in the space. *Problem Solver Techniques for Applied Computer Science, Com-S-477/577 Course Handout*, 3, 2020. 3

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[15] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 170–185, 2018. 7

[16] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 2

[17] Wojciech Matusik and Hanspeter Pfister. 3d tv: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. *ACM Transactions on Graphics (TOG)*, 23(3):814–824, 2004. 1

[18] 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. 2

[19] Dongkyung Nam, Jin-Ho Lee, Yang Ho Cho, Young Ju Jeong, Hyoseok Hwang, and Du Sik Park. Flat panel light-field 3-d display: concept, design, rendering, and calibration. *Proceedings of the IEEE*, 105(5):876–891, 2017. 2, 3

[20] Ken Perlin, Salvatore Paxia, and Joel S Kollin. An autostereoscopic display. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 319–326, 2000. 1

[21] 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, 4

[22] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9420–9429, 2024. 2

[23] Sheng Shen, Shujun Xing, Xinzhu Sang, Binbin Yan, and Yingying Chen. Virtual stereo content rendering technology review for light-field display. *Displays*, 76:102320, 2023. 1

[24] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3d photography using context-aware layered depth inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8028–8038, 2020. 3, 6, 7

[25] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[26] Hakan Urey, Kishore V Chellappan, Erdem Erden, and Phil Surman. State of the art in stereoscopic and autostereoscopic displays. *Proceedings of the IEEE*, 99(4):540–555, 2011. 1

[27] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer, 2025. 5, 7

[28] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[29] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022. 2

[30] Gaochang Wu, Belen Masia, Adrian Jarabo, Yuchen Zhang, Liangyong Wang, Qionghai Dai, Tianyou Chai, and Yebin Liu. Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):926–954, 2017. 1

[31] Dejia Xu, Weili Nie, Chao Liu, Sifei Liu, Jan Kautz, Zhangyang Wang, and Arash Vahdat. Camco: Camera-controllable 3d-consistent image-to-video generation. *arXiv preprint arXiv:2406.02509*, 2024. 3

[32] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10371–10381, 2024. 8

[33] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. 3, 6

[34] Zongyuan Yang, Baolin Liu, Yingde Song, Yongping Xiong, Lan Yi, Zhaohe Zhang, and Xunbo Yu. Directl: Efficient radiance fields rendering for 3d light field displays. *arXiv preprint arXiv:2407.14053*, 2024. 2, 3

[35] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 2

[36] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 5

[37] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024. 5, 6, 7

[38] Xunbo Yu, Xinzhu Sang, Duo Chen, Peng Wang, Xin Gao, Tianqi Zhao, Binbin Yan, Chongxiu Yu, Daxiong Xu, and Wenhua Dou. Autostereoscopic three-dimensional display with high dense views and the narrow structure pitch. *Chinese Optics Letters*, 12(6):060008, 2014. 2

[39] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018. 5