CVPR
#5987

7. Appendix

In this supplementary material, we provide detailed data processing methods and statistical details in Section A. Section B elaborates on the parameter settings for SIoU. Additionally, Section C presents supplementary experimental results.

A. Data Curation and Preprocessing.

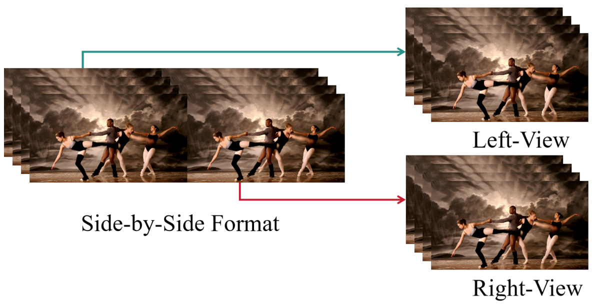

Figure 7. Input (Left-View) and ground truth (Right-View) images produced by splitting a video frame.

We collect a substantial amount of 3D content in the left-right format from movies and videos, as illustrated in Fig. 7. This format necessitates specific viewing equipment, such as 3D glasses, to ensure that the left and right eyes perceive the corresponding Left-View and Right-View images, respectively. By dividing these images from the middle, we create two distinct perspectives: the Leff-View and the Right-View images. Conversion to other stereoscopic formats can be achieved by applying appropriate processing techniques to this image pair.

```
# define label
texts_complexity = [
    "a simple scene with fewer than three objects",
    "a complex scene with many objects"]
texts_distance = [
    "an indoor scene",
    "an outdoor scene"]
```

Figure 8. CLIP scene categories.

For data statistics, we employ CLIP [34] as a scene classifier. Specifically, we feed text prompts and images into text encoder and image encoder of CLIP, respectively. We then calculate the cosine similarity between the resulting embeddings, assigning the category with the highest similarity as the classification result. Fig. 8 showcases the specific text prompts used. We perform pairwise statistics for the four categories (indoor, outdoor, simple, and complex). Additionally, we analyze the scene distribution within the dataset. As illustrated in Fig. 9, Mono2Stereo encompasses common indoor environments like living rooms and bedrooms, as well as more unique settings such as underwater scenes, cliffs, and rivers. For overall scene category statistics in Fig. 9, we utilize prompts in the format of “a/an [category] scene.”

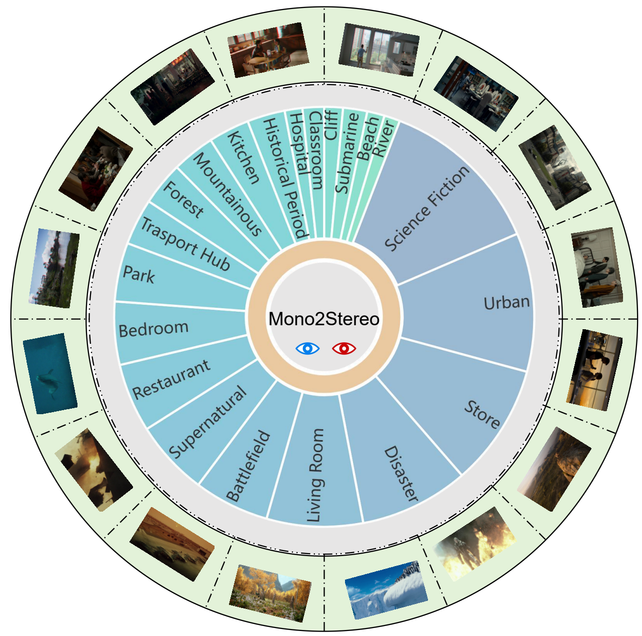

Figure 9. Distribution Characteristics of the Mono2Stereo Dataset.

B. Parameter Settings for SIoU.

To illustrate the individual roles of the two terms within SIoU, we conduct separate human subjective evaluations, with the results presented in Tab. 8. As shown, both terms contribute to achieving good consistency, suggesting that each reflects stereo quality to a certain extent by primarily focusing on the true disparity regions between the Left-View and Right-View images. Regarding the balancing parameter α in Eq. (1), we randomly divide 1100 image pairs into two sets: 500 pairs for optimal parameter and threshold searching, and 600 pairs for generalization validation. We experiment with various parameter settings for α, including 0.25, 0.5, 0.7, 0.75, and 0.8. Our findings indicate that these settings yield better consistency compared to a single item. Notably, $\alpha = 0.75$ demonstrates the highest level of consistency. Therefore, we set α to 0.75 for final SIoU. Validation on a set of 600 pairs, as illustrated in Tab. 8, shows no significant signs of overfitting.

For IoU2, employing a lower threshold ensures greater sensitivity to discrepancies, encompassing both disparity and pixel shifts. When validating across 500 sample pairs, we observe that a threshold of 5 yields the highest consistency. Attempting to decrease this threshold further actually

Table 8. Correlation with human judgements of stereo quality. IoU1 and IoU2 are components of the proposed SIoU metric. Both demonstrate correlation with human perception. Combining these components into SIoU yields even higher correlation scores. The results are based on a validation set of 600 pairs.

Metric	SIoU	IoU1	IoU2
Spearman Rank	**0.84**	0.81	0.80
Kendall Rank	**0.73**	0.70	0.68

Compute SIoU

```
def compute SIoU(left, right, pred):

    # Convert RGB images into gray sclae
    left_gray = RGB2Gray(left)
    right_gray = RGB2Gray(right)
    pred_gray = RGB2Gray(pred)

    # detect edges of the right and generated images
    right_edges = Canny(right_gray)
    pred_edges = Canny(pred_gray)

    # compute the differences
    diff_rl = abs(right_gray - left_gray)
    diff_gl = abs(pred_gray - left_gray)
    logical_rl = Zeroslike(diff_rl.shape)
    logical_gl = Zeroslike(diff_gl.shape)
    logical_rl[diff_rl > 5] = 1
    logical_gl[diff_gl > 5] = 1

    IoU1 = IoU(pred_edges, right_edges)
    IoU2 = IoU(logical_gl, logical_rl)
    SIoU = 0.75 * IoU1  + 0.25 * IoU2

    return SIoU
```

Figure 10. Pseudocode for the SIoU calculation process.

reduces consistency. This occurs because a lower threshold (< 5) incorporates more pixels into consideration, including those in areas that do not significantly impact the stereo effect, which is undesirable.

C. Supplementary Experimental Results.

C.1. Detailed Analysis of Two Conditions

In this paper, we define the complete Left-View image as the geometric condition, while the warped version of the Left-View image serves as the viewpoint condition. These conditions correspond to the inputs of single-stage and two-stage models, respectively. This section provides further clarification. As depicted in Fig. 11, the Left-View is a complete natural image, offering comprehensive geometric structure and texture details. Conversely, the Warped image, derived from the Left-View image through disparity warping, exhibits a perspective closer to the Right-View image. Therefore, the Left-View image provides richer geometric information, while the Warped image explicitly offers an observational viewpoint, spatially aligning it closer to the target. This distinction forms the basis for our naming convention and motivates our design of the dual-condition model, leveraging the complementary strengths of both conditions.

Figure 11. Visualization of Dual-Condition. The yellow circles highlight the differences in key spatial relationships, while the gray areas represent geometric differences.

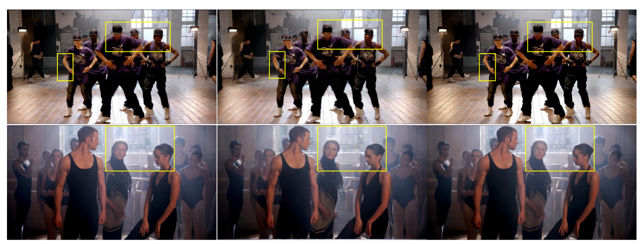

Figure 12. The influence of identical conditions on the output results. Areas with significant differences are highlighted by yellow boxes.

Furthermore, we present results under three different conditions, as illustrated in Fig. 12. Both the “Geometric” and “Viewpoint” conditions exhibit artifacts to varying degrees, with the “Viewpoint” condition displaying more pronounced artifacts due to its partially occluded input. In contrast, the “Dual-Condition” yields superior image quality.

C.2. Evaluating Performance in Various Scenes

To gain a deeper understanding of the performance across different scenarios, we evaluate models separately on five distinct scenes from the Mono2Stereo test dataset. As shown in Tab. 11, we observe that the model struggles in pairwise comparisons involving indoor, complex, and animation scenes. We hypothesize that this is due to limitations in three key areas where the model requires further improvement: disparity range estimation accuracy, geometric understanding, and color distribution handling. Consequently, we suggest that future research should focus on addressing these aspects. Finally, Mono2Stereo also provides 20 video clips for evaluating models. Despite our method being single-frame based, it still achieves promising results.

C.3. Why Velocity Edges?

Regarding the edge consistency constraint, the most intuitive approach appears to be constraining the edges within the latent space. Visualization of the feature maps, as illustrated in Fig. 13, confirms that both the latent and velocity exhibit positional correlation with the image. However, during training, we observe that predicting the latent or noise results in significantly slower convergence and even optimization failure, while velocity prediction does not suffer from these issues. Consequently, we opt to constrain the edges of the velocity field.

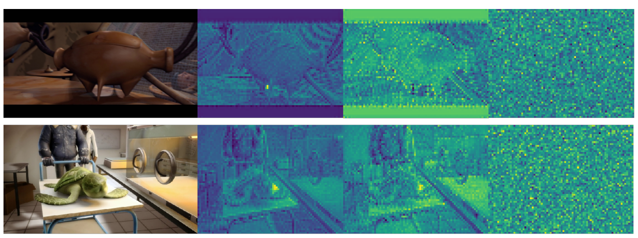

Image Latent Velocity Noise

Figure 13. Feature maps of latent and velocity.

C.4. Ablation Study on Inria 3DMovie.

To further validate the effectiveness of the Edge Consistency loss, we conduct out-of-domain performance evaluations using the Inria 3DMovie dataset, which comprises $2,727$ stereoscopic image pairs. As shown in Tab. 9, incorporating the Edge Consistency constraint consistently improves performance across all three tested conditions. This suggests that the benefits of this constraint are not limited to specific datasets, demonstrating its potential for generalization.

Table 9. Impact of LEC Loss across three conditions on Inria 3DMovie dataset.

Geo.	**View.**	**LEC Loss**	**Inria 3DMovie**			
			SIoU↑	RMSE↓	PSNR↑	SSIM↑
✓			0.2836	7.47	30.66	0.693
✓		✓	0.2949	7.46	30.68	0.693
	✓		0.3147	7.61	30.50	0.678
	✓	✓	0.3145	7.52	30.61	0.684
✓	✓		0.3147	7.44	30.70	0.691
✓	✓	✓	0.3186	7.31	30.85	0.697

Table 10. Impact of EC loss across three conditions. EC loss consistently improves performance, with notable gains in SIoU, the metric for perceived stereo quality.

LEC Loss	**Mono2Stereo**			
	SIoU↑	RMSE↓	PSNR↑	SSIM↑
0	0.2588	6.90	31.35	0.721
0.75	0.2608	6.83	31.45	**0.725**
1	**0.2619**	**6.82**	**31.45**	0.721
1.25	0.2615	6.88	31.38	0.719

C.5. Ablation Study on Edge Consistency Loss

When applying the Edge Consistency loss, we conduct experiments to validate the impact of different α values in Eq. (3) within a small range. Using the dual-condition diffusion model, we experiment with α values of 0.75, 1, and 1.25, while $\alpha = 0$ represents the absence of the edge constraint. As Tab. 10 illustrates, applying the edge consistency constraint at varying strengths consistently leads to improvements in SIoU, indicating that the constraint term is not overly sensitive to the specific α value. We offer an additional analysis: when α is 0, all pixels in the image are optimized equally. The edge constraint, in essence, imposes a stricter penalty on regions that genuinely influence

Table 11. Evaluating models across various scenes.

Method	Indoor		Outdoor		Complex		Simple		Animation		Video	
	SIoU↑	RMSE↓	SIoU↑	RMSE↓	SIoU↑	RMSE↓	SIoU↑	RMSE↓	SIoU↑	RMSE↓	SIoU↑	RMSE↓
StereoDiffusion [46]	0.2387	7.48	0.2441	7.68	0.2182	7.78	0.2571	6.17	0.2296	8.01	0.1992	8.38
Geometric Condition	0.2505	5.31	0.2543	5.74	0.2561	5.94	0.2791	4.28	0.2525	5.73	0.2610	5.61
Viewpoint Condition	0.2761	5.71	0.2824	6.02	0.2713	6.62	0.2986	5.76	0.2764	6.49	0.2735	5.95
Dual Condition	0.2819	5.21	0.2969	5.65	0.2894	5.78	0.3095	4.29	0.2999	5.76	0.2817	5.50