UniPercept: A Unified Diffusion Model for Generalizable Visual Perception

Supplementary Material

7. Additional Ablation Studies

7.1. Effect of Inference Steps

We employ the second-order DPM-Flow-Solver [77] for inference. As shown in the first block of Tab. 12, we evaluate the effect of the number of inference steps by running 1 (first-order), 2, 6, 10, and 20-step sampling on the surface normal estimation task without ensemble inference. Across these settings, fewer inference steps consistently yield better quantitative performance. In contrast, as illustrated in Fig. 8, using more inference steps generally produces outputs with richer fine-grained details. To balance inference speed and visual details, we adopt 6-step sampling for all qualitative visualizations.

7.2. Effect of Ensemble Sampling

In the second block of Tab. 12, we evaluate the influence of ensemble sampling by averaging predictions obtained from six runs with different random noise. After ensembling, the quantitative results across different step settings become more comparable, as multi-step sampling benefits more from the ensemble procedure, whereas the gain for the 1-step setting remains limited.

Table 12. Quantitative comparison of different inference settings on normal estimation on the Scannet dataset. NFEs [35] is the number of function evaluations required to obtain the prediction, i.e., *ensemble* $\times$ *steps*.

NFEs	mean↓	11.25° ↑	30° ↑
1 × 20	19.9	50.2	79.4
1 × 10	19.6	50.9	80.0
1 × 6	19.1	51.8	80.6
1 × 2	17.4	55.1	82.5
1 × 1	**17.0**	**55.7**	**82.8**
6 × 20	17.3	54.9	82.0
6 × 10	17.2	55.3	82.2
6 × 6	17.0	55.6	82.5
6 × 2	**16.8**	**56.1**	**83.0**
6 × 1	16.9	55.7	**83.0**

7.3. Effect of DINOv3

Single-task We first evaluate the effect of DINOv3 [70] in single-task setting. Following the same training strategy used for our foundation model, we independently train depth and normal estimation models for 10K steps each. As shown in Table 13 and Table 14, incorporating DINOv3 improves performance on both perception tasks.

Table 13. Effect of DINOv3 on the NYUv2 dataset for depth estimation (single-task).

	AbsRel↓	δ_1 ↑	δ_2 ↑
w/o DINOv3	8.4	92.7	98.5
w/ DINOv3	**8.2**	**92.9**	**98.7**

Table 14. Effect of DINOv3 on the NYUv2 dataset for normal estimation (single-task).

	mean↓	11.25° ↑	30° ↑
w/o DINOv3	18.5	49.3	80.3
w/ DINOv3	**18.4**	**50.0**	**80.4**

Multi-task Next, we evaluate the effect of DINOv3 in a multi-task setting, where the model is trained on a set of seven base tasks as in our foundation setup. As shown in Tab. 15, the improvements introduced by DINOv3 are limited in the multi-task scenario compared to single-task training. This is likely due to the foundation model has already acquired strong common representations across different perception tasks, diminishing the additional benefit provided by DINOv3. Despite this, DINOv3 still offers modest improvements.

Table 15. Effect of DINOv3 on the NYUv2 dataset (multi-task).

	Depth			Normal		
	AbsRel↓	δ_1 ↑	δ_2 ↑	mean↓	11.25° ↑	30° ↑
w/o DINOv3	9.2	91.5	**98.6**	**20.4**	41.4	**78.4**
w/ DINOv3	**9.1**	**91.6**	**98.6**	**20.4**	**42.6**	78.1

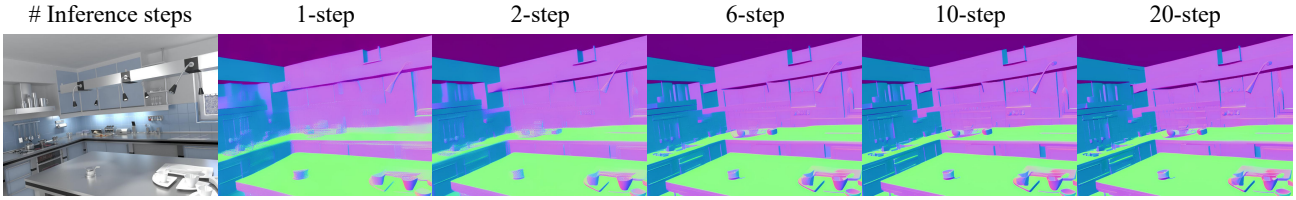

Figure 8. Visual comparison of different inference steps.

Table 16. Effect of the fine-tuning methods.

Methods	Depth		Normal		Albedo		Irradiance		Edge		DIS	
	AbsRel ↓	δ_1 ↑	mean ↓	11.25° ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	ODS ↑	OIS ↑	F^x_β ↑	S_m ↑
LoRA	8.32	92.8	19.0	**50.4**	10.65	0.523	14.41	0.644	0.730	0.737	0.728	0.762
Adapter (ours)	**8.18**	**93.2**	**18.6**	49.8	**11.63**	**0.537**	**14.92**	**0.657**	**0.773**	**0.793**	**0.752**	**0.784**

7.4. Effect of Finetune Methods

Tab. 16 presents a comparison between LoRA and our adapter-based fine-tuning strategy, evaluating both performance on base tasks and adaptation to novel tasks. Specifically, we first pretrain UniPercept for 20K steps on the seven base tasks, and then fine-tune it for an additional 20K steps on novel tasks (edge and DIS). Both methods are configured with comparable parameter counts (12.73M for our adapter with a bottleneck ratio of 64, and 12.90M for LoRA with rank 72), ensuring a fair comparison under the same lightweight tuning budget. Across nearly all evaluated tasks, our adapter achieves superior performance. These results indicate that our adapter design enables more stable and effective task adaptation than LoRA under the same lightweight fine-tuning setting.

7.5. Effect of Timestep Schedule

In this section, we validate the effectiveness of the proposed **half-logit-normal** timestep schedule and compare it with the original logit-normal sampling distribution, as shown in Fig. 9. The key difference between the two methods is that our proposed timestep schedule places more emphasis on sampling in the high-noise phases rather than in the middle stage of the denoising process. This design choice is based on the intention to better support few-step inference. In this setting, the model can take advantage of the information contained in the image input to obtain accurate results during the early stages of denoising.

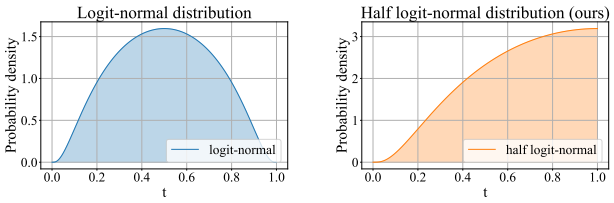

Figure 9. Comparison of Timestep Sampling Distributions.

Quantitative experiments, summarized in Tab. 17, were conducted by training the base seven tasks for 10K steps using both timestep schedules. We evaluate the performance on depth and normal estimation tasks using the Scannet dataset. The results show that the proposed schedule improves performance over the logit-normal in both tasks.

Table 17. Effect of the timestep schedule on the Scannet dataset.

	Depth			Normal		
	AbsRel↓	δ_1 ↑	δ_2 ↑	mean↓	11.25° ↑	30° ↑
logit-normal	10.6	88.2	97.7	21.9	37.9	75.8
half-logit-normal (ours)	**10.3**	**88.7**	**97.9**	**20.8**	**38.4**	**78.2**

8. Additional Visual Results

We provide additional visual results for various tasks that UniPercept can handle, as shown in Figs. 10 to 23.

9. Notes on Datasets

For the semantic segmentation task, we follow Jodi [80] and group the original 150 ADE20K classes into 12 superclasses to improve color separability in RGB space, at the cost of reduced label diversity.

For the albedo and irradiance tasks, we follow Marigold [35] and filter out invalid samples in the Hypersim dataset [67] by verifying the relation

$$\text{color} \approx (\text{reflectance} \times \text{illumination}) + \text{residual}, \quad (7)$$

and discarding samples whose reconstruction error exceeds the PSNR threshold of 40.

Figure 10. Additional visual results of depth estimation.

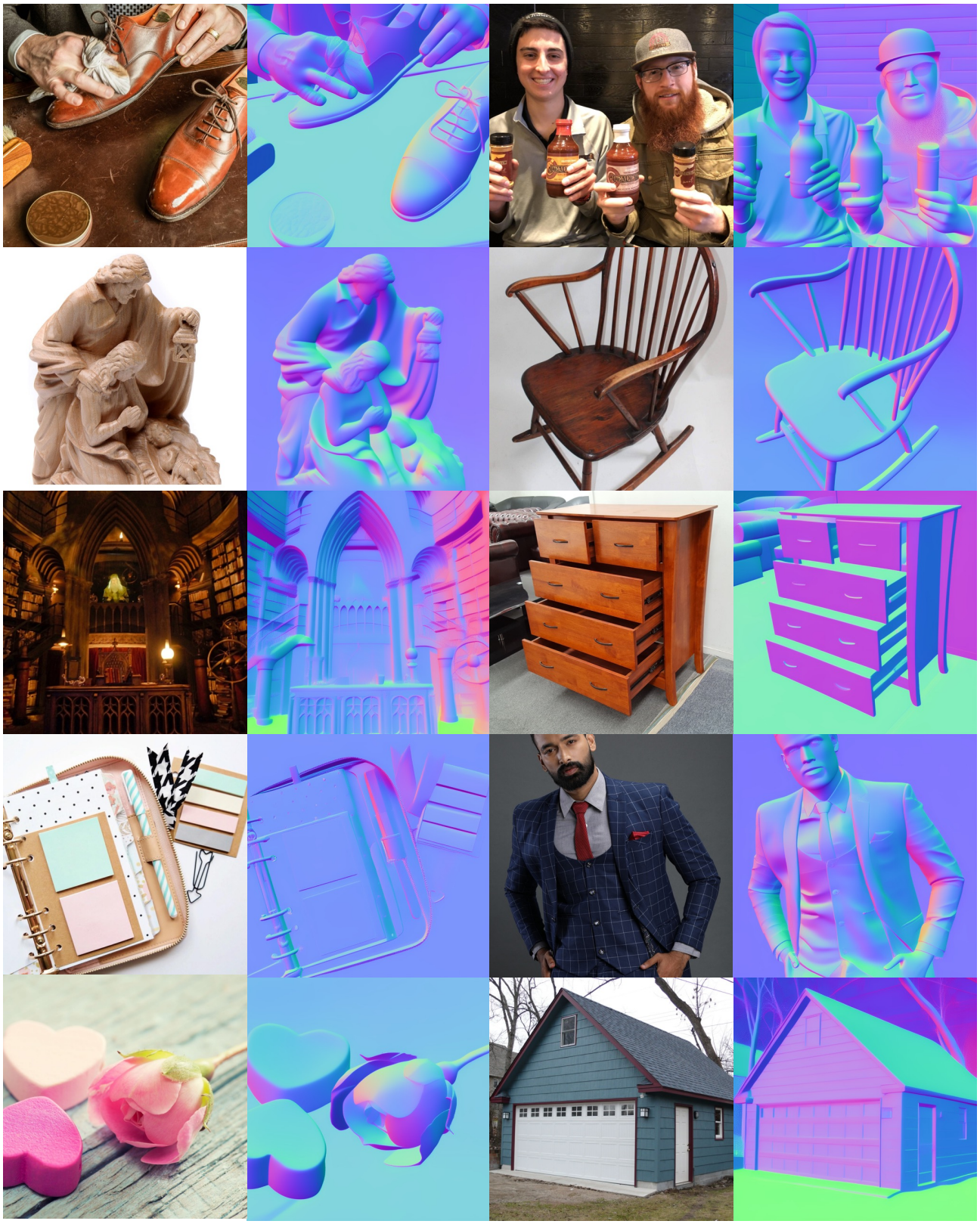

Figure 11. Additional visual results of normal estimation.

Figure 12. Additional visual results of albedo estimation.

Figure 13. Additional visual results of irradiance estimation.

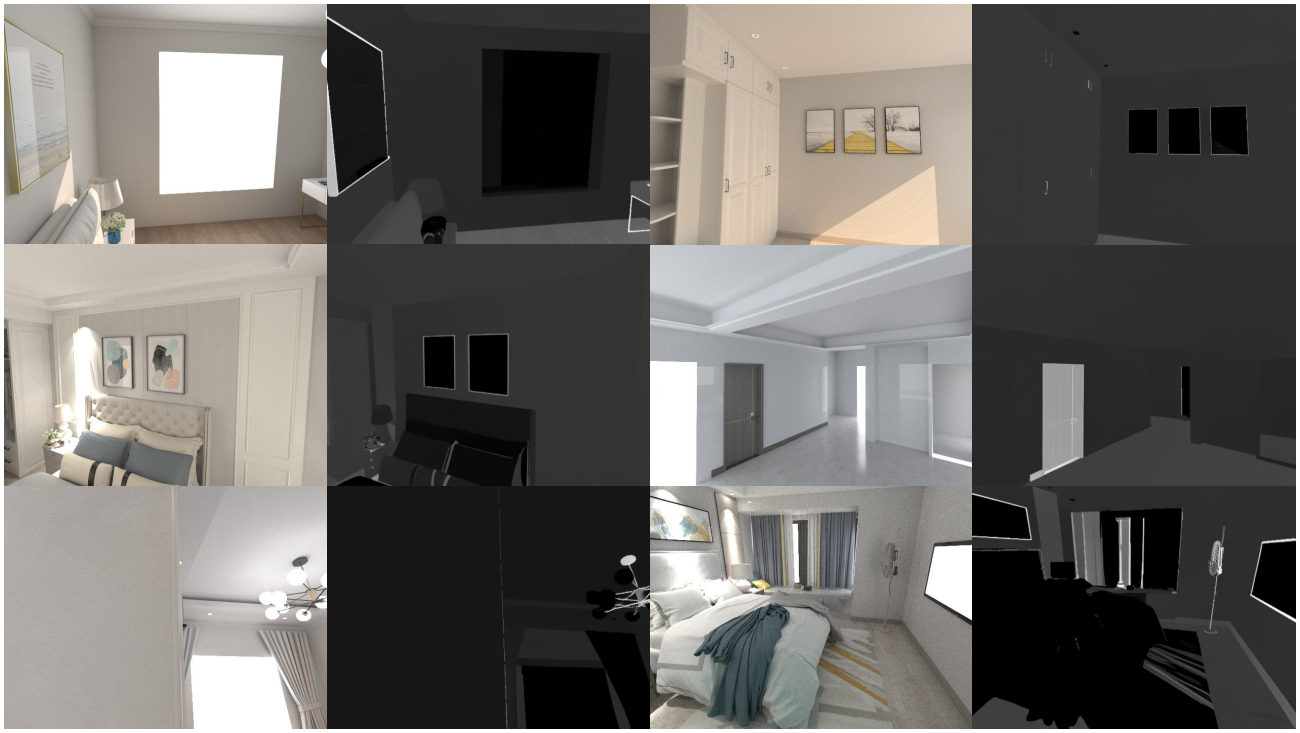

Figure 14. Additional visual results of metallicity estimation.

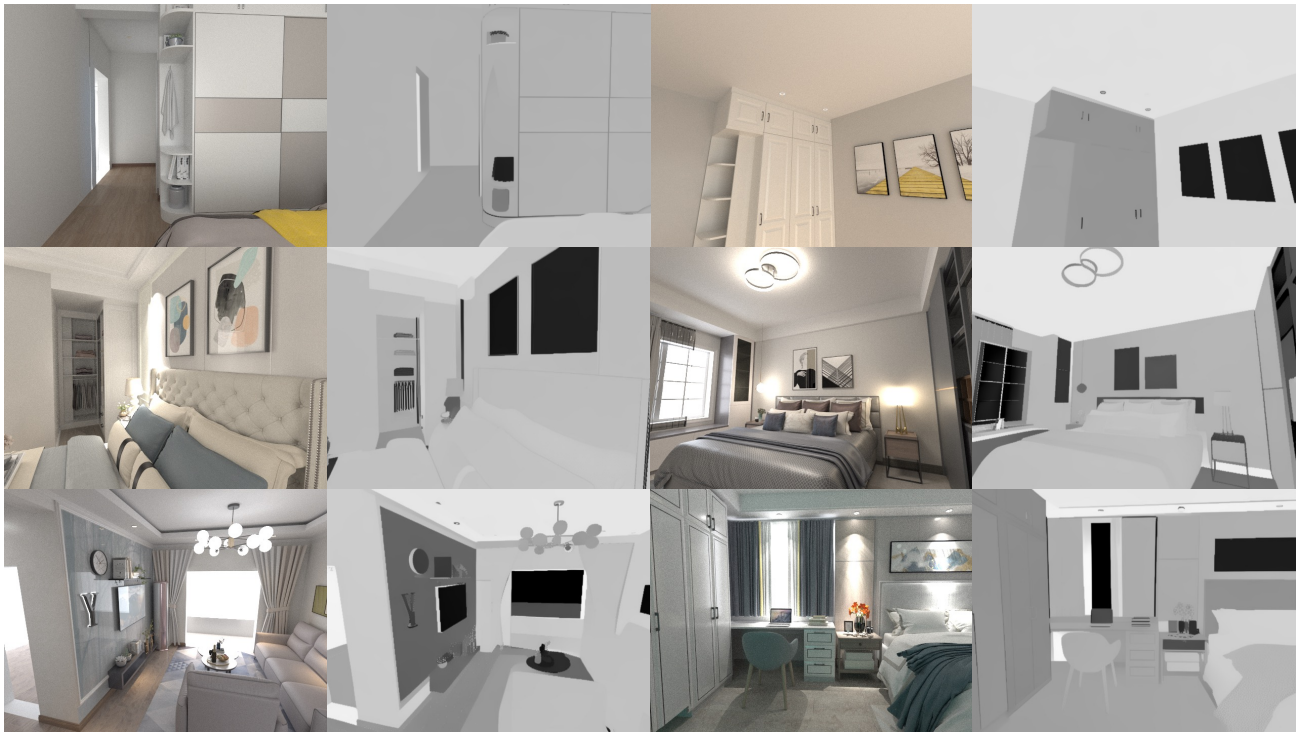

Figure 15. Additional visual results of roughness estimation.

Figure 16. Additional visual results of semantic segmentation.

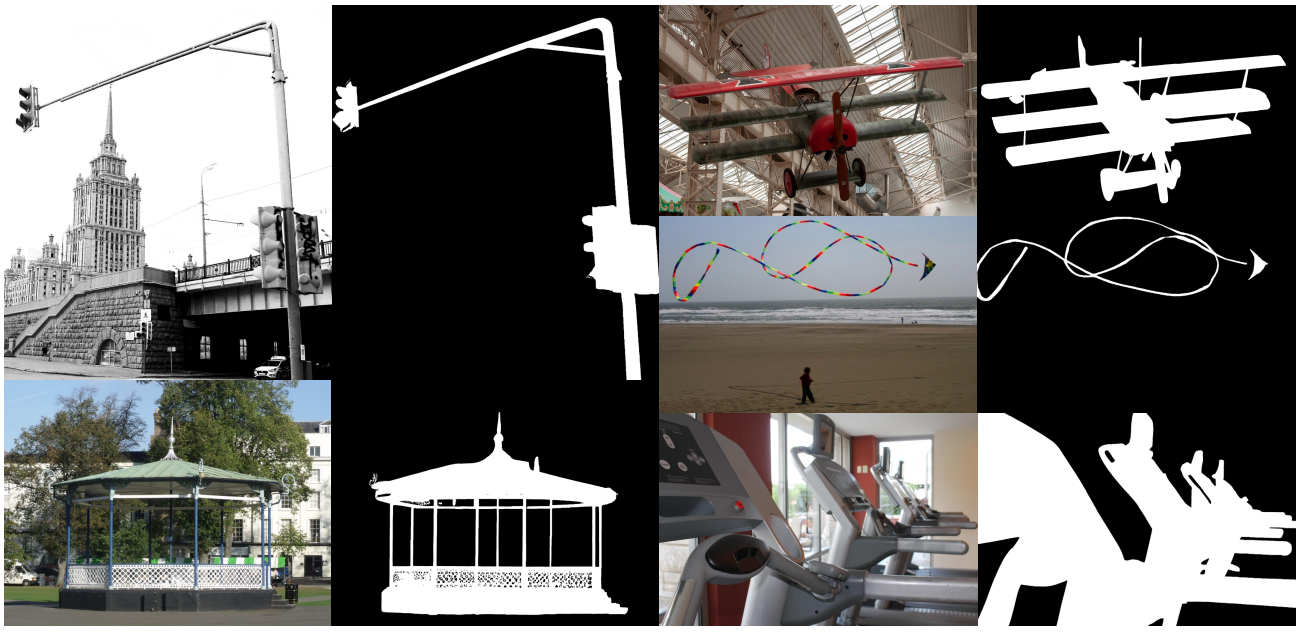

Figure 17. Additional visual results of dichotomous segmentation.

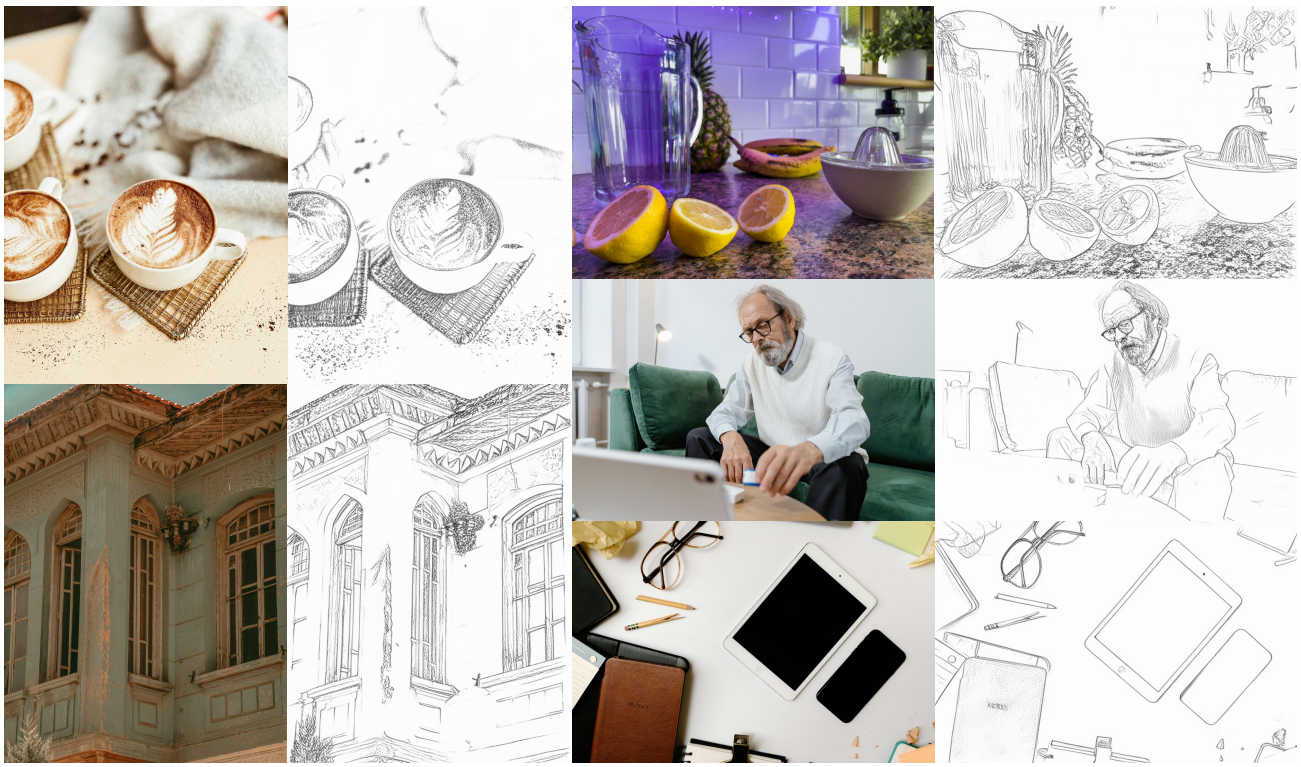

Figure 18. Additional visual results of line art estimation.

Figure 19. Additional visual results of edge detection.

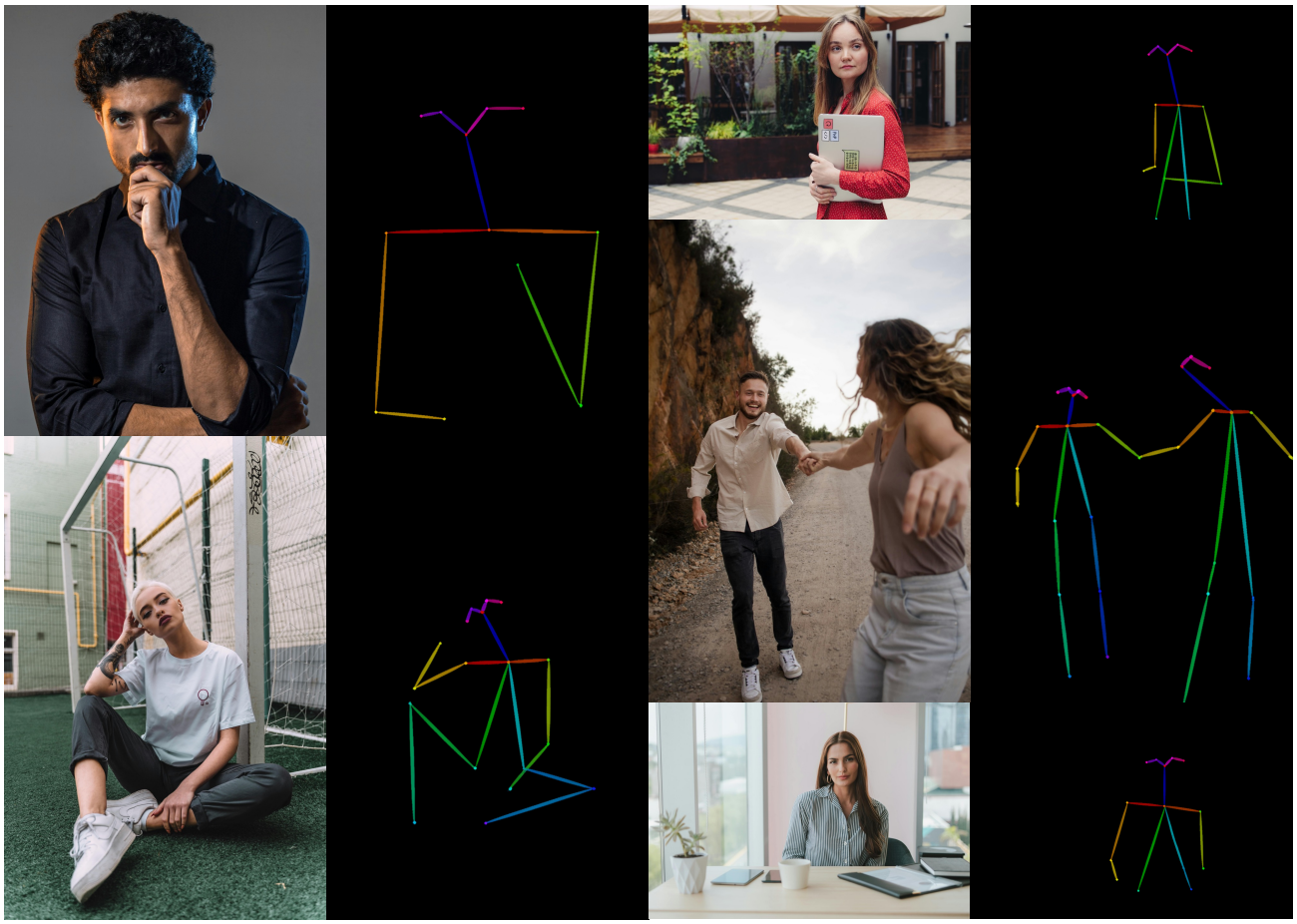

Figure 20. Additional visual results of human skeleton detection.

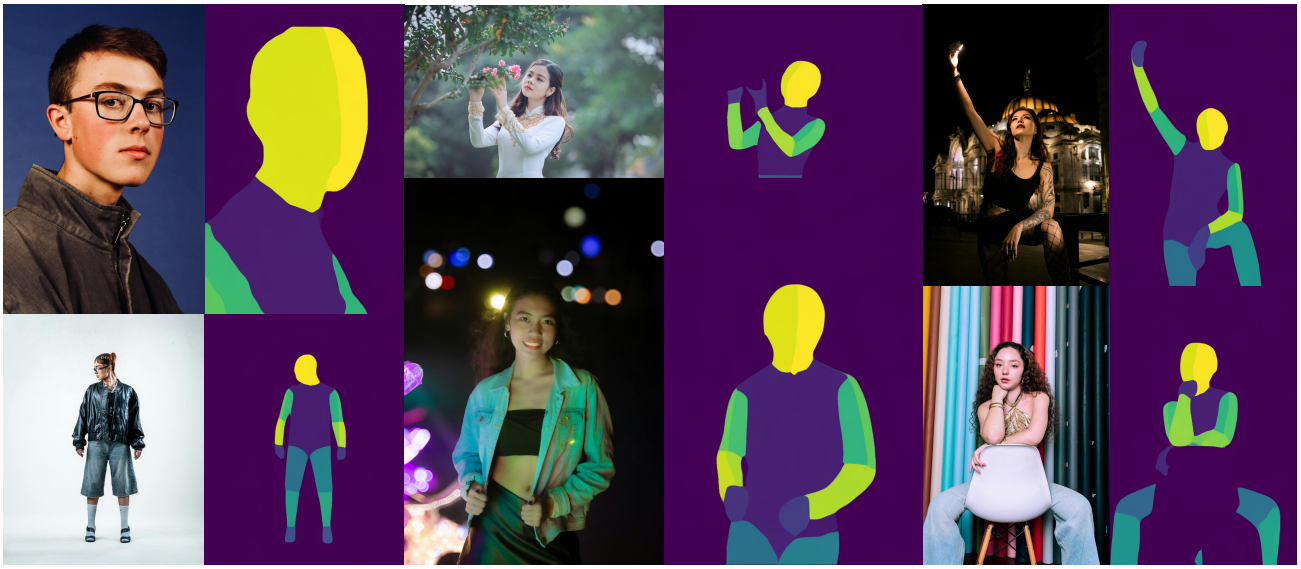

Figure 21. Additional visual results of densepose detection.

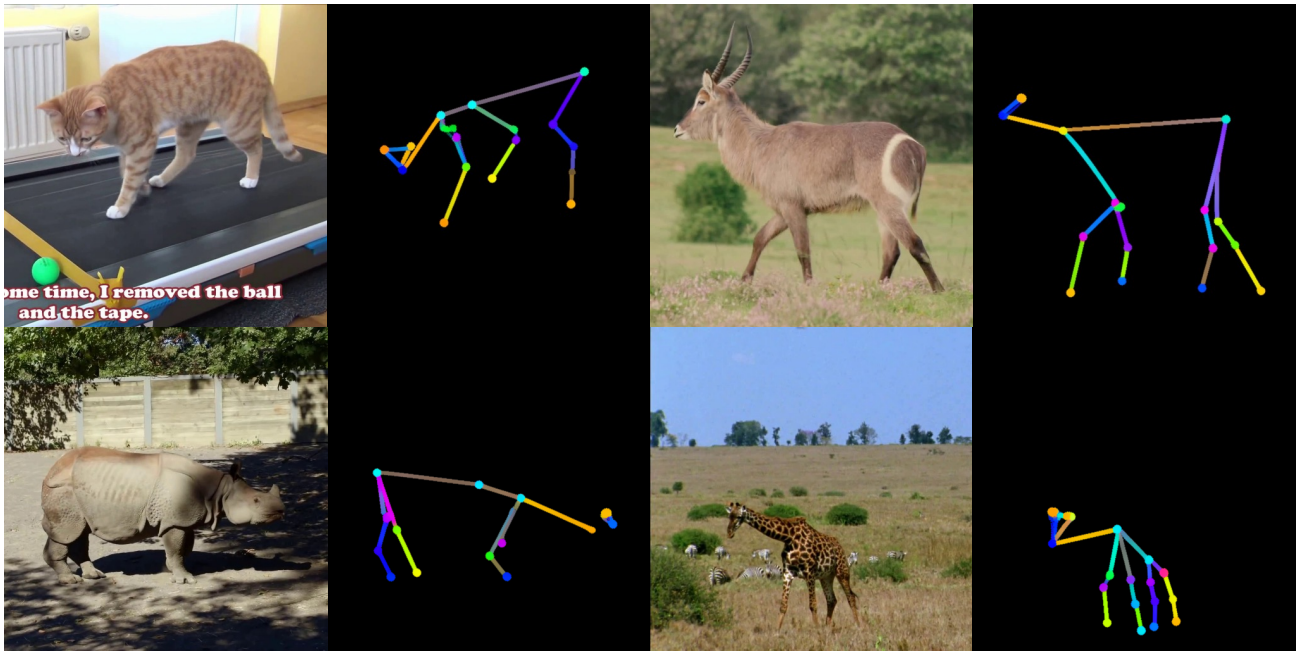

Figure 22. Additional visual results of animal pose detection.

Figure 23. Additional visual results of line segment detection.