		b3-scene		b10-scene		b20-scene		b50-scene		100-ball		1k-ball	
		time(s)	PSNR	time	PSNR	time	PSNR	time	PSNR	time	PSNR	time	PSNR
wo s	rs	0.240	25.78	0.658	23.32	1.678	20.02	1.881	18.67	1.887	26.54	16.63	25.26
	wo rs	0.209	23.97	0.443	19.92	0.873	15.26	1.209	15.43	1.741	24.19	16.34	22.58
shadow	rs	0.398	\	1.040	\	2.441	\	2.943	\	3.073	\	28.97	\
	wo rs	0.367		0.813		1.628		2.286		3.002		27.66	

Table 2: Performance of the implemented Pytorch framework on an N-object dataset ('wo s' means no shadow is calculated; 'rs' stands for resampling, i.e., outlier processing in Table 1). The resolution is set to 900×600. We use the naive composition (right side in Fig. 6) by NeRF as ground truth for composition without shadows. We only test the average time (in seconds) in each frame for shadow composition. It should be noted that the quality of the 1K-ball pressure testing was superior to the former due to the simpler geometry of the objects. Nonetheless, due to excessive overlapping between balls, the time increases rapidly.

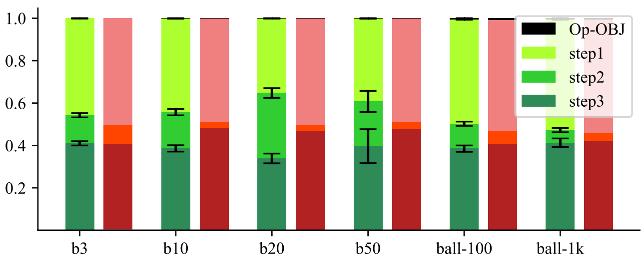

Figure 8: The time consumption ratio of our framework for the main steps. *Green* and *Red* bars represent the results with and without outliers resampling, respectively. Regarding the green bar, refer to this figure in conjunction with Fig. 6, when the camera view varies, certain objects which occupy a larger portion of the frame, e.g., the trees, have a higher resampling ratio and thus increase the processing time. Focusing attention on the red bar, the Step 2 (deferred shading step) consistently assumes a relatively minor temporal allocation across all scenarios. This is because, regardless of the number of objects in the scene, a pixel will be shaded only once with a single NeRF object.

virtual scene in a progressive and interactive manner. The real-time manipulation benefits from a buffer reuse technique to accelerate the performance during manipulation. Specifically, when the user manipulates an object, the other objects' rendering processes are frozen, and their buffers are reused to decrease the computational load.

Compatibility with traditional graphics pipelines

The proposed framework is also suitable for integration with traditional rendering pipelines to build a mixed pipeline that supports dynamic shadow casting across meshes and neural objects in a single scene (see Fig. 7). Specifically, we first use a shared camera & light settings across mesh and neural scene to generate rendering buffers. Subsequently, these two scenes are seamlessly integrated using depth-based composition. Finally, shadow mapping is applied to produced a mixed result.

Conclusions and Limitations

We present a general framework for a fast composition of NeRF objects with dynamic shadows. The core technical contribution is the use of Neural Depth Fields to allow for the direct intersection of rays and implicit surfaces to quickly determine the spatial relationship between objects as well as surface color computation of NeRF objects. To the best of our knowledge, this is the first framework that allows for the rapid creation of a virtual scene composed of a large number of NeRF objects with dynamic shadows. Our method can also be combined with traditional renderers to form a mixed pipeline, increasing its versatility and applicability.

The proposed framework is currently limited to solid surface objects, and predicting weight distribution in local space rather than a single depth may be a better option. In the future, this compositing framework can be further enhanced with more powerful editable NeRF works (**???????**) to achieve global illumination and improved visual quality.

Acknowledgments

This work was supported by Key R&D Program of Zhejiang (No. 2023C01047) and the National Natural Science Foundation of China (Grant No. 62036010). The authors acknowledge the support of Bytedance's MM Lab's GPU cluster, as well as Yanzhen Chen's early assistance in searching for various works.

Aperiam aliquid quam placeat laboriosam tempora velit quaerat aliquam, rem fugiat minus amet odio cum, sunt numquam praesentium corrupti culpa eveniet aspernatur molestiae dicta dolor ad.Nostrum aliquam blanditiis in deserunt assumenda harum alias id, fugiat libero eligendi.Minus facere placeat et explicabo sit vel inventore optio consequatur aliquid, dignissimos consequatur fuga dolorem maiores recusandae minima amet provident, eligendi quasi mollitia ducimus dolores vel, at nulla nihil cum ducimus labore dignissimos aspernatur?Soluta placeat quam sunt odio, necessitatibus natus libero distinctio illum.Minus ipsam nisi vero, necessitatibus ex rerum magnam illum numquam quis cumque aut, nesciunt error quisquam odit distinctio quod porro nostrum ea cum veritatis esse, officiis doloremque nesciunt possimus sit repudiandae libero, molestias quidem nesciunt non magnam ducimus porro aliquam esse beatae?Voluptates iusto animi ducimus eum et quasi doloremque quaerat voluptatem, eius eveniet suscipit consectetur nam omnis ullam assumenda, in