Multimodal Neural Surface Reconstruction: Recovering the Geometry and Appearance of 3D Scenes from Events and Grayscale Images

Sazan Mahbub **Brandon Y. Feng** **Christopher A. Metzler**
Department of Computer Science
University of Maryland
College Park, MD 20742
{smahbub, yfeng97, metzler}@cs.umd.edu

Abstract

Event cameras offer high frame rates, minimal motion blur, and excellent dynamic range. As a result they excel at reconstructing the geometry of 3D scenes. However, their measurements do not contain absolute intensity information, which can make accurately reconstructing the appearance of 3D scenes from events challenging. In this work, we develop a multimodal neural 3D scene reconstruction framework that simultaneously reconstructs scene geometry from events and scene appearance from grayscale images. Our framework—which is based on neural surface representations, as opposed to the neural radiance fields used in previous works—is able to reconstruct both the structure and appearance of 3D scenes more accurately than existing unimodal reconstruction methods.

1 Introduction

View synthesis in 3D scenes has long captivated computer vision research. The introduction of neural radiance fields (NeRFs) and their success in crafting life-like 3D scene representations have spurred numerous subsequent studies. NeRFs excel in encoding continuous signals via neural networks, obviating the need for discrete sample storage and processing. They can generate novel scene views from arbitrary camera poses and viewpoints. In comparison to conventional methods like Structure-from-Motion [1] and light-field photography [2], NeRF and its successors, empowered by neural fields and neural volume rendering, have achieved remarkable results in 3D scene reconstruction [3–6]. These advancements have left a profound impact across diverse industries, spanning robotics, urban mapping, augmented and virtual reality, and the entertainment sector [7].

Conventional camera images face limitations in low-light conditions, long exposure times, motion blur, and a limited dynamic range. Event cameras address these limitations by offering motion blur resistance, high dynamic range, high temporal resolution, and low power consumption. They find applications in object tracking [8, 9], optical flow estimation [10, 11], and geometry reconstruction [12, 13].

Recent studies demonstrated 3D reconstruction using neural radiance fields and event cameras [14–16]. However, event-camera-based approaches have their own challenges. Firstly, the novel views they generate often lack realism and quality, failing to align with human perception, while conventional-camera-based methods produce more convincing results [3, 14–17]. Secondly, event-camera-based methods often rely on external sensors for pose estimation, adding costs to experiments or real-world deployments, whereas conventional camera based methods handle both training and pose estimation without the need for extra sensors [3, 17]. Lastly, event-camera-based methods usually require significantly longer training times compared to conventional cameras.

NeurIPS 2023 Workshop on Deep Learning and Inverse Problems.

The efficacy of neural surface reconstruction methods (such as NeuS, NeuS-2, and NDR [5, 18, 19]) over NeRFs has been highlighted recently, especially when trained on RGB and RGB-D data. While NeRF-based methods dominate event-based 3D scene reconstruction [14–16], the potential of NeuS-like methods in this context remains unexplored.

In this study, we propose a new framework to effectively learn neural scene representation on data from two different modalities – event camera and conventional grayscale camera. Particularly, we disentangle the learning of geometry and visual appearance by guiding their training with events and grayscale images, respectively. Additionally, we explore the potential of neural surface reconstruction from events for the first-time. We empirically demonstrate the capability of our framework to leverage the benefits of both data modalities and side-step their shortcomings.

2 Method

We establish a pipeline to train NeuS[5] using both *intensity* (from conventional grayscale cameras) and *accumulated event-frames* (from event cameras), alongside a similar pipeline for NeRF (Neural Radiance Field)[3] for experimentation and comparison. Accumulated events represent cumulative events over time at each 2D coordinate on the view plane [14]. First, we define the neural volume rendering pipeline and the forward function we use to simulated events. Next, we introduce a new training approach to efficiently learn from both data modalities where we decouple the learning of geometry and visual appearance. Specifically, we use event-frames to guide 3D geometry learning and intensity (grayscale-frames) for learning visual appearance.

2.1 Neural volume rendering

We aim to learn a function $f(x, y, z, d)$ from that, given a 3D coordinate (x, y, z) and a viewing direction d, can output the opaque density and intensity of that point through neural surface reconstruction similar to [5]. Then, we can render 2D images from a particular viewing direction using volume rendering.

Our method starts with sampling N points $\{(x_i, y_i, z_i) \mid 0 \leq i \leq N\}$ along the viewing direction d of a ray r. Their densities $\{\sigma_i \mid 0 \leq i \leq N\}$ and intensities $\{I_i \mid 0 \leq i \leq N\}$ are computed using the function $f(x_i, y_i, z_i, d)$. These are aggregated to compute the rendered intensity $\mathcal{I}(r, t) = \sum_{i=i}^{N} T_i \alpha_i I_i$. Here, T_i denotes the accumulated transmittance, indicating the likelihood of ray r reaching the i-th coordinate, while α_i represents the opacity of this point, as discussed in previous works [3–5, 18, 19]. It's worth noting that NeuS [5] and NeRF [3] employ different sets of volume rendering equations to compute T_i and α_i. Detailed information can be found in their respective papers. In our approach, we adopt NeuS's formulation, where the learnable function is represented using two consecutive feed-forward neural networks, namely the "SDF network" ($SDFnet_\theta$) and the "intensity network" ($Inet_\phi$), parameterized by θ and ϕ respectively.

However, it is important to note that, unlike previous approaches, we apply the *softplus activation* function at the end of $Inet_\phi$ instead of the sigmoid. The softplus function allows for a range of $[0, \infty]$, which better corresponds to the brightness levels a camera sensor can capture, making it more suitable for event simulation, as shown in [20].

2.2 Forward function to simulate events

An event camera records asynchronous brightness changes at any pixel (u, v) as a stream of events, $\{e_i(u, v, t_i)\}_{i=1}^{N_e}$, with each event at timestamp t_i is $e_i = p(u, v, t_i)\,|C(u, v, t_i)|$, where $p_i(u, v, t_i) \in \{+1, -1\}$ is the event's polarity and $C(u, v, t_i)$ is a hardware dependant threshold. An event-generation roughly follows the following inequality,

$$|\log(\mathcal{I}_{tr}(u, v, t_i + \delta t)) - \log(\mathcal{I}_{tr}(u, v, t_i))| \geq |C(u, v, t_i)|, \tag{1}$$

where $\mathcal{I}_{tr}$ is the true intensity at pixel (u, v) and time t_i, and δt is the time elapsed. Based on this, accumulated event $\Delta\mathcal{E}$ can be computed as,

$$\Delta\mathcal{E}(u, v, \delta t) = \Sigma_{t_i \in \delta t}\ e(u, v, t_i) = \Sigma_{t_i \in \delta t}\ p(u, v, t_i)\,|C(u, v, t_i)|. \tag{2}$$

Now, we can approximate $\Delta\mathcal{E}$ as the *change in measured brightness*, $\Delta L(u, v, \delta d)$, using the rendered intensities $\mathcal{I}(r, t)$ and $\mathcal{I}(r, t + \delta t)$,

$$\Delta\mathcal{E}(u, v, \delta t) \approx \Delta L(u, v, \delta t) = CM(\mathcal{I}(r, t + \delta t)) - CM(\mathcal{I}(r, t)), \tag{3}$$

where r represents a ray through the point (u, v), and $CM(.)$ is a function that approximately maps

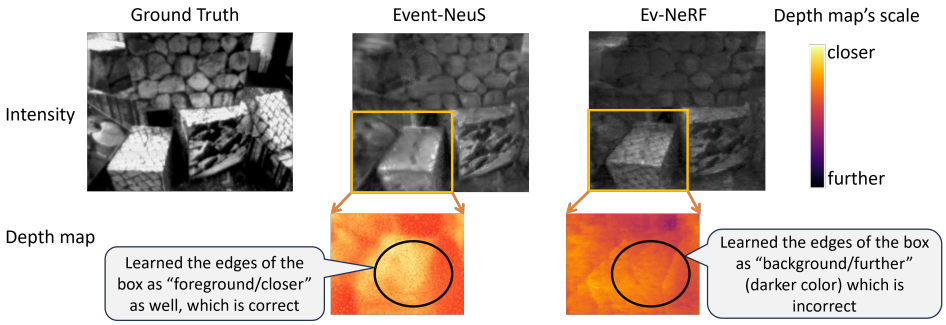

Figure 1: Comparison between Event-NeuS and Ev-NeRF. **Top:** gamma-corrected intensity. **Bottom:** depth maps. Surface-based reconstructions more accurately recover 3D geometry.

the rendered intensity $\mathcal{I}(r,t)$ to measured brightness by the camera. We note that, in reality, the camera measurement does not always follow a linear or logarithmic relation with the actual intensity as shown in Eq. 1. In fact, it is mostly linear until a certain threshold and after that it approximately follows a logarithmic function, as shown by Hu *et al.* [20]. We approximate this function as,

$$CM(\mathcal{I}) \approx \begin{cases} \log_{\mathcal{B}}(\mathcal{I}), & \text{if } \mathcal{I} > \mathcal{B}, \\ \mathcal{I}/\mathcal{B}, & \text{otherwise}, \end{cases} \tag{4}$$

where $\mathcal{B}$ serves as the base of log when $\mathcal{I} > \mathcal{B}$, otherwise as a scaling factor. This formulation ensures a smooth transition from linear to log scale, analogous to event-camera sensors [20].

2.3 Our proposed training strategy for multimodal learning

Our main idea is to disentangle the geometry and appearance training processes. We learn the 3D geometry *only using events*, since event cameras can capture details *even in extreme low-light condition*, which conventional cameras cannot. We use the grayscale images only to optimize the intensity so that it improves the appearance of 2D projections but cannot hurt the accuracy of learnt geometry from events.

To learn the geometry, we leverage the event-rendering loss proposed by [15] (Eq. 5) for its robustness against event noise. They assume the event threshold $C(\cdot)$ in Eq. 2 to be fairly constant across space, which gave them promising accuracy. It is represented as two learnable functions for positive and negative events, $C^+(t)$ and $C^-(t)$.

$$\mathcal{J}_e = \begin{cases} \|\Delta\mathcal{L} - \Delta\mathcal{E} - C^+\|_2^2, & \text{if } \Delta\mathcal{L} - \Delta\mathcal{E} > C^+, \\ \|\Delta\mathcal{L} - \Delta\mathcal{E} - C^-\|_2^2, & \text{if } \Delta\mathcal{L} - \Delta\mathcal{E} < C^-, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

To prevent $C^+(t)$ and $C^-(t)$ from trivially producing zeros, [15] also formulated a constraint as $\mathcal{J}_{et} = \Sigma_t ReLU(C_0^+ - C^+(t)) + ReLU(C^-(t) - C_0^-)$. Here, C_0^- and C_0^+ are predefined minimum and maximum possible values of $C^-(t)$ and $C^+(t)$, respectively. We train both $SDFnet_\theta$ and $Inet_\phi$ on event signals, minimizing the total "event-loss" $\mathcal{J}_{event}^{\theta,\phi} = \mathcal{J}_e + \lambda_e \mathcal{J}_{et}$, where λ_e is a predefined scaling factor.

With a motivation to improve perceptual accuracy of rendered images, we also minimize the L_1-loss between rendered intensity $\mathcal{I}(\cdot)$ and true intensity $\mathcal{I}^*(\cdot)$ from the conventional grayscale camera. However, $\mathcal{I}^*(\cdot)$ may differ from the event camera's intensity range. To handle this, we learn a scaling factor η for the normalized true intensity, $\tilde{\mathcal{I}}^*(\cdot) = normalize(\mathcal{I}^*(\cdot)) \in [0,1]$, and minimize the loss $\mathcal{J}_\mathcal{I} = \Sigma_{r,t} \|\eta\tilde{\mathcal{I}}^*(r,t) - \mathcal{I}(r,t)\|_1$. We note that without any constraints on η, we can end up with $\eta = 0$ and a trivial solution of $\mathcal{I}(r,t) = 0$. To prevent this, we include a regularization term $\mathcal{J}_{\mathcal{I}t} = ReLU(\eta_{min} - \eta)$ to prevent η from going below η_{min}, a predefined constant. We also employ the Eikonal term $\mathcal{J}_{eik}$ for our NeuS-based framework to regularize the signed-distance function, as proposed in [5]. Our final "intensity-loss" is $\mathcal{J}_{intensity}^{\phi} = \mathcal{J}_\mathcal{I} + \lambda_\mathcal{I}\mathcal{J}_{\mathcal{I}t} + \lambda_{eik}\mathcal{J}_{eik}$, where $\lambda_\mathcal{I}$ and λ_{eik} are two predefined weighting factors. We use $\mathcal{J}_{intensity}^{\phi}$ only to optimize the parameters ϕ in our intensity network $Inet_\phi$ so that it can improve the perceptual accuracy, but cannot affect the learnt geometry from events, as discussed before.

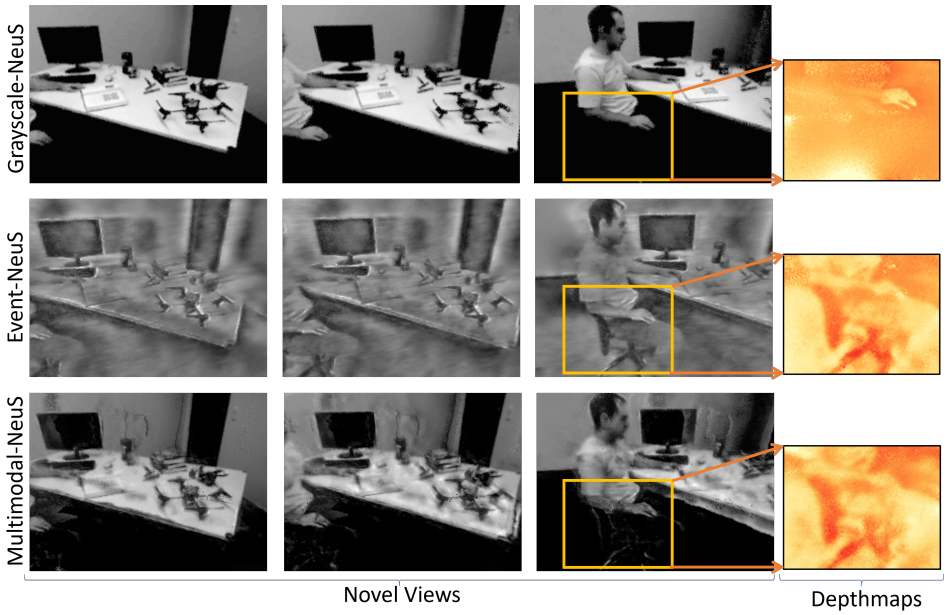

Figure 2: Rendered novel views along with the depth-maps for a region with *extreme low-light condition*. **Top:** Grayscale-NeuS achieved good perceptual accuracy but failed to learn the geometry in that region. **Mid:** Event-NeuS learned the geometric detail better, but has low perceptually accuracy. **Bottom:** Our proposed Multimodal-NeuS method can learn the geometric detail and also achieve good perceptual accuracy.

3 Results and Discussion

For our experiments we use two of the datasets curated by [21]: boxes-6dof and dynamic-6dof. First, we compare the performance of NeuS and NeRF trained on event camera data, with Ev-NeRF [15] as a representative of the event-based NeRF methods [14–16]. Here, we term the NeuS trained on events as Event-NeuS. Figure 1 displays a sample reconstruction. Although Event-NeuS does not seem to produce better visual appearance (intensity), we can see some evidence of better underlying geometry, compared to Ev-NeRF.

In Figure 2, we demonstrate the rendered grayscale images from three settings, along with the depth-maps for a particular region of interest in the scene (*low-light region under the table*). Please note that in the ground-truth grayscale images, this region is completely dark with no detail available.

Here we can see that the Grayscale-NeuS cannot seem to reconstruct the detailed geometry of that region (shown in depth-map); however, its generates grayscale frames are realistic looking with good perceptual accuracy. On the other hand, the depth-map for Event-NeuS shows that it could learn the geometric detail in that region pretty well. Despite this, its rendered images are not realistic looking (e.g., it totally failed to represent that the table-top and the monitor should be the bright- and dark-colored, respectively). Moreover, rendered images by Event-NeuS exhibit noise and reduced sharpness, potentially due to sparsity and noise in events. The last row shows that our proposed Multimodal-NeuS can utilize the best of both worlds – its rendered novels views have perceptual accuracy close to Grayscale-NeuS, and it can still learn the geometric details of the scene as good as the Event-NeuS.

4 Conclusion

In this study we develop and apply a novel, multimodal approach for reconstructing 3D scenes from events and grayscale images. Our framework yields promising results in terms of both 3D geometry reconstruction and novel-view rendering. In the future, we plan to extend our method to reconstruct dynamic scenes.

Acknowledgements

S.M., B.F., and C.M. were supported in part by the AFOSR Young Investigator Program Award FA9550-22-1-0208.

References

[1] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] Yan Zhou, Huiwen Guo, Ruiqing Fu, Guoyuan Liang, Can Wang, and Xinyu Wu. 3d reconstruction based on light field information. In *2015 IEEE International Conference on Information and Automation*, pages 976–981, 2015.

[3] 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.

[4] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.

[5] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems*, 34:27171–27183, 2021.

[6] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022.

[7] Kyle Gao, Yina Gao, Hongjie He, Denning Lu, Linlin Xu, and Jonathan Li. Nerf: Neural radiance field in 3d vision, a comprehensive review. *arXiv preprint arXiv:2210.00379*, 2022.

[8] Anton Mitrokhin, Cornelia Fermüller, Chethan Parameshwara, and Yiannis Aloimonos. Event-based moving object detection and tracking. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9, 2018.

[9] Ju Hong Yoon, Chang-Ryeol Lee, Ming-Hsuan Yang, and Kuk-Jin Yoon. Online multi-object tracking via structural constraint event aggregation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[10] Guillermo Gallego, Henri Rebecq, and Davide Scaramuzza. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[11] Patrick Bardow, Andrew J. Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[12] Henri Rebecq, Timo Horstschäfer, Guillermo Gallego, and Davide Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robotics and Automation Letters*, 2(2):593–600, 2016.

[13] Hanme Kim, Stefan Leutenegger, and Andrew J Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, pages 349–364. Springer, 2016.

[14] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. Eventnerf: Neural radiance fields from a single colour event camera. *arXiv preprint arXiv:2206.11896*, 2022.

[15] Inwoo Hwang, Junho Kim, and Young Min Kim. Ev-nerf: Event based neural radiance field. *arXiv preprint arXiv:2206.12455*, 2022.

[16] Simon Klenk, Lukas Koestler, Davide Scaramuzza, and Daniel Cremers. E-nerf: Neural radiance fields from a moving event camera. *arXiv preprint arXiv:2208.11300*, 2022.

[17] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021.

[18] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. *arXiv preprint arXiv:2212.05231*, 2022.

[19] Hongrui Cai, Wanquan Feng, Xuetao Feng, Yan Wang, and Juyong Zhang. Neural surface reconstruction of dynamic scenes with monocular rgb-d camera. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 967–981. Curran Associates, Inc., 2022.

[20] Tobi Delbruck, Yuhuang Hu, and Zhe He. V2e: From video frames to realistic dvs event camera streams. *arXiv e-prints*, pages arXiv–2006, 2020.

[21] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International Journal of Robotics Research*, 36(2):142–149, 2017.

[22] Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. Fastnerf: High-fidelity neural rendering at 200fps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14346–14355, 2021.

[23] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7467–7477, 2020.

[24] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7210–7219, 2021.

[25] Christian Reiser, Songyou Peng, Yiyi Liao, and Andreas Geiger. Kilonerf: Speeding up neural radiance fields with thousands of tiny mlps. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14335–14345, 2021.

[26] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5752–5761, 2021.

[27] Alexis Baudron, Zihao W Wang, Oliver Cossairt, and Aggelos K Katsaggelos. E3d: Event-based 3d shape reconstruction. *arXiv preprint arXiv:2012.05214*, 2020.

[28] Henri Rebecq, Guillermo Gallego, Elias Mueggler, and Davide Scaramuzza. Emvs: Event-based multi-view stereo—3d reconstruction with an event camera in real-time. *International Journal of Computer Vision*, 126(12):1394–1414, 2018.

[29] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 235–251, 2018.

A Additional Related Work

In this section we review additional related work.

Neural 3D scene reconstruction using conventional cameras. Neural Field approaches have gained widespread recognition for their exceptional 3D scene reconstruction capabilities from 2D RGB and RGB-D data [3, 4, 6, 22]. NeRF's pioneering work [3] achieved groundbreaking results by integrating neural 3D fields with neural volume rendering [23] to learn 3D geometry from images with known positions and camera directions. NeRF++[4] addressed rendering ambiguities and expanded it to handle ambiguities and 360-degree unbounded scenes. [24] focused on 3D reconstruction in the presence of transient objects and varying luminance. Some studies developed methods to speed up training and inference of NeRFs, e.g., FastNeRF[22], InstantNGP [6]. FastNeRF[22] optimized neural radiance fields for mobile devices and mixed reality, reducing computation time. KiloNeRF [25], proposed by Reiser et al., employed numerous small multi-layer perceptrons to accelerate NeRF by learning different scene parts. InstantNGP [6], introduced by M"ulller et al., incorporated hashing to handle high-resolution images, achieving a significant speed-up in NeRF training. Martin-Brualla et al. presented an approach [24] for accurate 3D geometry reconstruction in the presence of transient objects and varying luminance. Yu et al. [26] achieved an astounding

3000-fold speed-up in NeRF training through the use of PlenOctrees. Recent research also explores neural surface reconstruction, employing signed distance function, from RGB images [5, 18] and RGB-D images [19], often surpassing several NeRF-like approaches in performance.

Neural 3D scene reconstruction using event cameras. Event cameras are a high potential for 3D reconstruction, especially when conventional cameras fail to capture enough detail due to low-light conditions [14–16, 20, 27–29]. Event-camera measurements contain the information of the time-derivative of log-intensity, quantized by a threshold that is a hardware-dependent function of space, time, and polarity of intensity-change(+1 for increment and -1 for decrement in intensity) [16, 20]. E3D [27] enforces 3D mesh consistency with neural rendering. Zhou et al.[29] and Rebecq et al.[28] propose 3D reconstruction for multi-view stereo event cameras. Recent studies explore 3D volume reconstruction from event-camera data, using neural representation learning and rendering [14–16]. EventNeRF [14] by Rudnev et al. is supervised on events accumulated over short and long periods. Ev-NeRF [15] offers improved robustness againts event noise and generating coherent 3D structures. Klenk et al. introduce E-NeRF [16], designed for high-motion event cameras in 3D scene representation.