

EventNeuS: 3D Mesh Reconstruction from a Single Event Camera

Supplementary Material

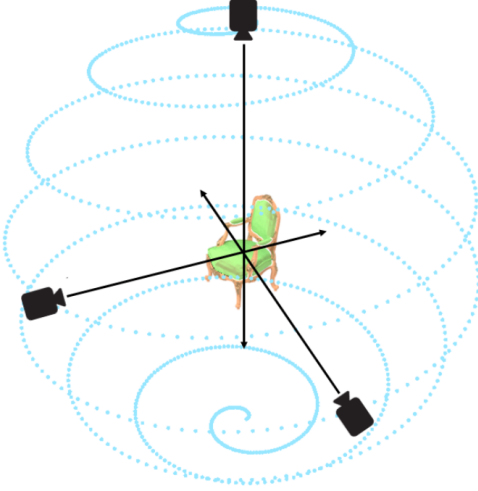


Figure 7. Visualisation of Seiffert’s spherical spiral trajectory of the virtual monocular event camera used in our synthetically generated data. See Sec. 5.1.1 and App. A for further details.

This supplementary document provides additional details on the datasets (App. A), implementation details (App. B), and more visualisations of the qualitative experimental results (App. C).

A. Synthetic Data Generation

We generate synthetic event streams using ground truth meshes from the NeRF [19] synthetic datasets. The 3D assets are imported into Blender’s Cycles renderer, where we render 999 uniformly sampled views along Seiffert’s spherical spiral trajectory, Fig. 7. Our capture sequence comprises eight complete revolutions around each object, beginning at the base orientation and ascending to the apex. Each RGB frame is rendered at 346×260 pixels resolution with HDR lighting to simulate realistic intensity variations.

Event streams are synthesised using ESIM [30], an event camera simulator that computes per-pixel logarithmic brightness changes. We configure ESIM with the following parameters:

```
"Resolution": "260, 346"  
"bg_val": "159.0/255.0"  
"THR": "0.2"  
"gamma": "2.2"
```

B. Implementation Details

SH Encoding. To model view-dependent colour variations efficiently, we encode camera directions \mathbf{d} using TinyCudaNN’s [23] optimised SH implementation. This produces a compact 16D feature vector. For a direction \mathbf{d} , we compute:

$$\text{SH}(\mathbf{d}) = \text{tcnn.SphericalHarmonics}(\mathbf{d}) \in \mathbb{R}^{16}$$

For camera directions \mathbf{d} , we configure the encoder with:

```
encoding_config = {  
    "otype": "SphericalHarmonics",  
    "degree": 4,  
}
```

These 16D SH bases are further concatenated with surface normals $\nabla\varphi(\mathbf{x})$ and geometric features from the SDF network $\varphi(\mathbf{x})$ to predict view-dependent colours.

Evaluation Metrics. To compute the SDF-MAE, we implement a hybrid sampling strategy: 50% of the evaluation points are sampled directly from the mesh surfaces with slight random perturbations (within $\pm 1\%$ of the mesh’s maximum dimension) to capture fine surface details, while the remaining 50% are uniformly sampled within a bounding box that encloses both meshes. For each sampled point, we calculate the absolute difference between the ground truth and predicted SDF values.

To compute CD, both meshes are first normalised to a unit cube to ensure scale invariance. We then uniformly sample 10^4 points from each mesh’s surface and perform nearest-neighbour search using a KD-Tree. This bidirectional metric penalises both missing and extraneous geometry, effectively capturing discrepancies in surface alignment.

C. Additional Visualisations

We provide comprehensive evaluations across standard NeRF synthetic scenes. As quantitatively demonstrated in Table 1, EventNeuS achieves superior reconstruction quality, particularly evident in geometrically complex structures like the Hotdog’s sausage, where our method preserves thin protrusions and surface curvature with 0.0839 Chamfer distance versus 0.1024-0.4281 for baselines. Qualitative comparisons (Fig. 8) further reveal that EventNeuS preserves more surface details than any other baseline.

C.1. Novel-View Synthesis Evaluation

While our work primarily focuses on geometry reconstruction, we additionally evaluate the novel view synthesis capabilities of EventNeuS. We compute PSNR, SSIM,

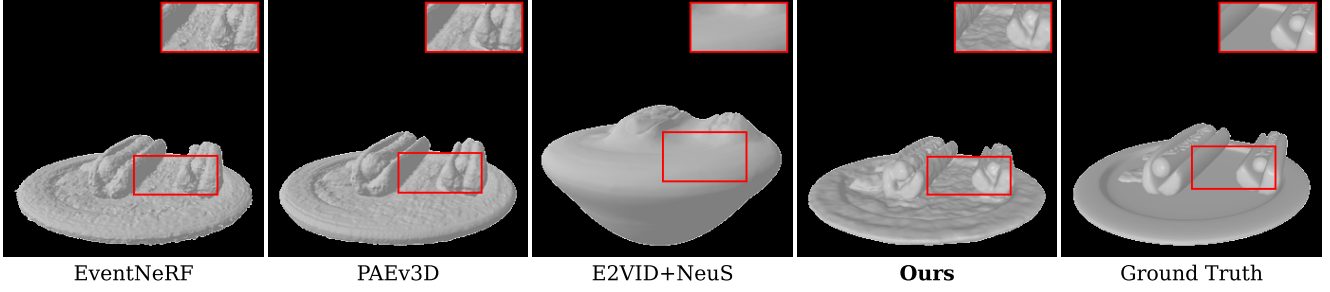


Figure 8. Qualitative comparison on the Hotdog scene. E2VID+NeuS [28, 35] produces oversmoothed geometry that loses fine structural details. The results of EventNeRF [30] and PAEv3D [34] both suffer from noticeable surface jitter artefacts, particularly evident along the sausage curvature, where the density-based representations fail to recover smooth surface. Our EventNeuS produces a significantly cleaner reconstruction, accurately preserving the surface smoothness and characteristic shape of the hotdog with fewer artefacts.

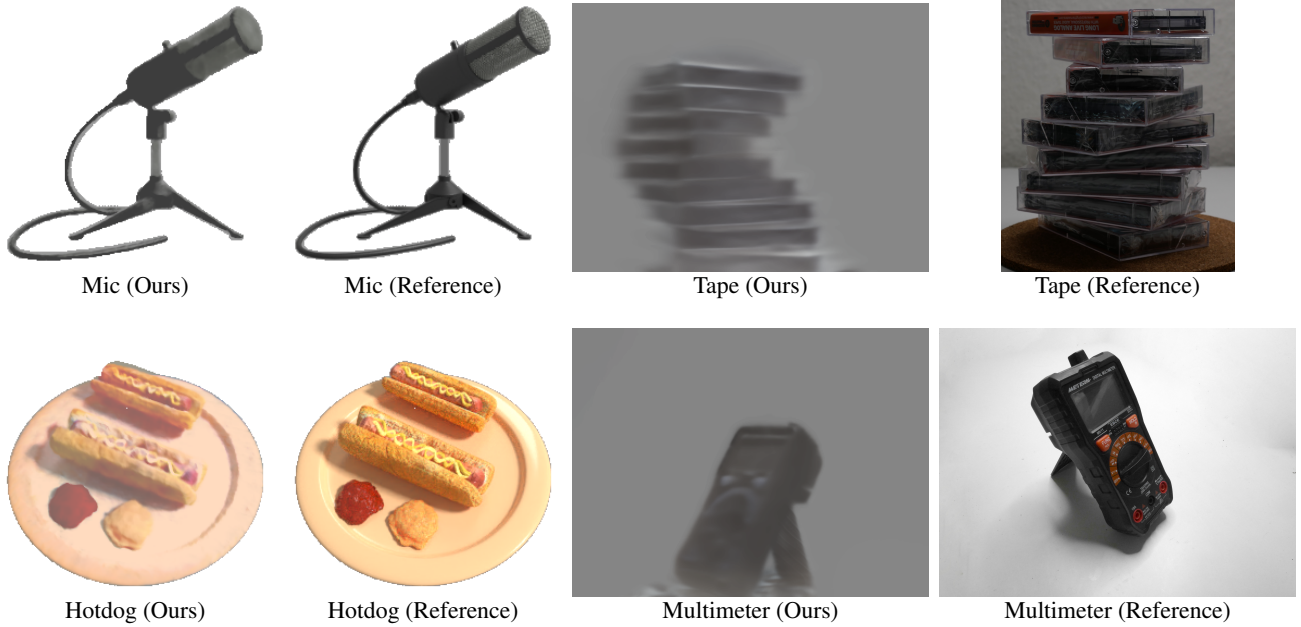


Figure 9. Novel-view synthesis results demonstrating EventNeuS’s capability to render high-quality views from learned geometry, showcasing both synthetic and real-world scenarios. Reference RGB images are shown for reference only and are not used during training.

and LPIPS metrics on regions of interest (ROIs) using ground-truth foreground masks for both EventNeuS and EventNeRF. As shown in Tab. 2 (main paper), EventNeuS achieves competitive performance across all metrics, with particularly notable improvements in SSIM and LPIPS. An important conclusion from these results is that better geometry facilitates more accurate novel views, as evidenced by our consistent improvements in perceptual quality metrics. Fig. 9 shows qualitative examples of our novel view synthesis results on both synthetic and real data, demonstrating EventNeuS’s capability to render high-quality views from the learned geometry.

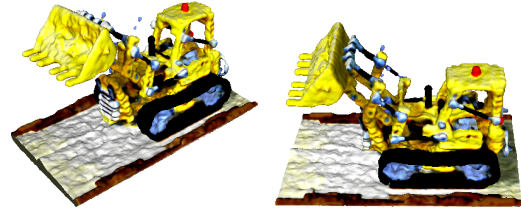


Figure 10. Textured mesh of the Lego scene obtained using our EventNeuS approach, shown from two arbitrary viewpoints.

C.2. Coloured Meshes

Once EventNeuS is trained, we extract the final mesh by querying both the SDF network f_{sdf} and the colour field f_{colour} on a dense grid. We then apply the Marching Cubes algorithm [15] to generate vertices from the SDF grid, while simultaneously interpolating vertex colours from the colour grid. The interpolation is weighted by the inverse SDF values to prioritise points closer to the surface. This process yields a high-fidelity textured mesh, as demonstrated for the Lego scene in Fig. 10. The rendered views exhibit sharp geometric features and consistent colouration, such as the crisp edges and uniform surfaces of the Lego bricks. This result underscores our method’s capability to jointly reconstruct precise geometry and detailed appearance solely from an event stream.