EventNeuS: 3D Mesh Reconstruction from a Single Event Camera

Shreyas Sachan[1,2] Viktor Rudnev[1,2] Mohamed Elgharib[2] Christian Theobalt[2] Vladislav Golyanik[2]

[1]Saarland University, SIC [2]MPI for Informatics, SIC

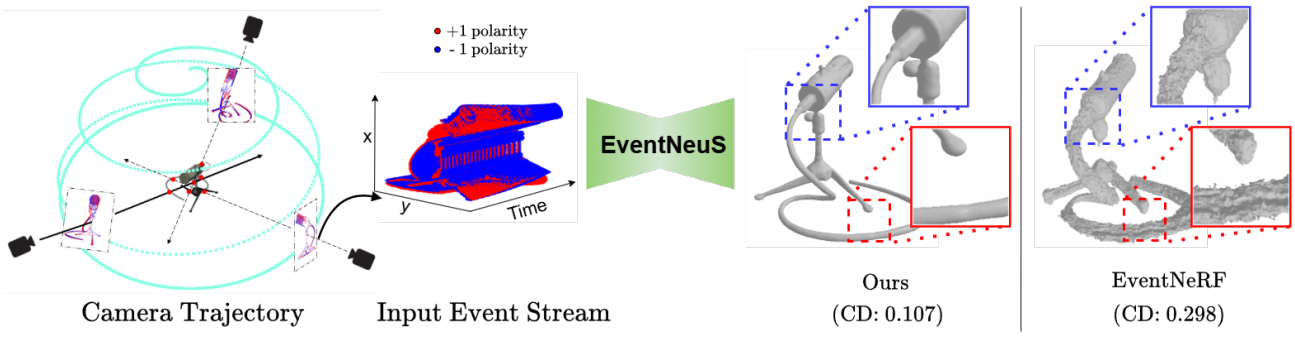

Figure 1. Left: A moving monocular event camera capturing the asynchronous per-pixel brightness changes, with red (positive) and blue (negative) polarities representing pixel intensity changes over time. Centre: The captured event stream is processed by the proposed **EventNeuS** to reconstruct a detailed 3D mesh. Right: Compared to the previous state of the art [30], our method produces a more accurate reconstruction with a lower Chamfer Distance (CD: 0.107 vs 0.298), highlighting its effectiveness in the event-based 3D reconstruction.

Abstract

Event cameras offer a considerable alternative to RGB cameras in many scenarios. While there are recent works on event-based novel-view synthesis, dense 3D mesh reconstruction remains scarcely explored and existing event-based techniques are severely limited in their 3D reconstruction accuracy. To address this limitation, we present EventNeuS, a self-supervised neural model for learning 3D representations from monocular colour event streams. Our approach, for the first time, combines 3D signed distance function and density field learning with event-based supervision. Furthermore, we introduce spherical harmonics encodings into our model for enhanced handling of view-dependent effects. EventNeuS outperforms existing approaches by a significant margin, achieving 34% lower Chamfer distance and 31% lower mean absolute error on average compared to the best previous method.

1. Introduction

Most existing methods for 3D reconstruction rely on RGB inputs and techniques such as Structure-from-Motion [6, 13, 33] and Multi-View Stereo [31, 32, 41] for depth estimation and surface reconstruction. While the current state of the art achieves high 3D reconstruction accuracy, it often struggles with rapid motion or varying lighting conditions, leading to motion blur and inaccuracies in 3D reconstruction. To address these limitations, event-based methods have been recently explored, leveraging the high temporal resolution and dynamic range of event cameras [2]. While event cameras offer advantages in high-speed settings, existing event-based 3D reconstruction techniques result in sparse representations and cannot reconstruct dense scene surface details. Moreover, many of these methods depend on explicit feature matching [18, 42], which is challenging considering the sparse and asynchronous nature of event data. It can further limit the reconstruction accuracy or require additional synchronous RGB recordings, potentially diminishing the utility of event observations and the multimodal setup overall. Hence, there remains a need for approaches that can fully exploit the unique capabilities of event cameras to achieve dense and accurate 3D surface reconstruction, particularly methods that operate exclusively with event data without the compromises often required in multimodal setups.

This work proposes EventNeuS, a new method for detailed surface reconstruction solely from monocular event streams. We capitalise on the insights from the recent literature on event-based novel-view synthesis [30] and surface extraction from neural radiance fields [25, 35]. EventNeuS leverages a neural implicit Signed Distance Function (SDF) [26] to implicitly model scene geometry using per-colour-channel changes captured by a single event camera. It accu-

mulates events over time in event frames, effectively capturing the dynamic changes occurring within the scene, as illustrated in Fig. 1. The event frames supervise a coordinate-based neural network that learns an SDF of the observed scene. The SDF maps any 3D point of the reconstructed volume to its distance from the nearest surface, enabling high-fidelity surface extraction upon convergence.

We introduce several innovations compared to previous techniques. First—to improve the modelling of view-dependent effects—we replace conventional positional encoding for camera view directions with Spherical Harmonics (SH) encodings, applying them in event-based 3D reconstruction for the first time. Second, our novel self-supervised architecture enables joint learning of a neural radiance field (NeRF) [19] and an SDF of a scene relying solely on camera-pose-informed event-based supervision. We design a loss function that aligns temporal differences in the rendered RGB images with the accumulated event data, effectively capturing the per-pixel changes over time and enabling neural network weight optimisation. Additionally, we incorporate regularisation terms to ensure smoothness and consistency across the reconstructed surfaces. In summary, our technical contributions are as follows:

1) EventNeuS, the first method to extract high-quality 3D meshes **solely** from monocular event streams, with no RGB frames or explicit feature matching, enabling dense surface reconstruction (Sec. 4);
2) Novel integration of **spherical harmonics encoding, hierarchical importance sampling, and frequency-annealing technique** for capturing view-dependent effects and learning fine geometric details from sparse event data (Sec. 4.3);
3) A new **spiral-trajectory synthetic dataset for 3D mesh reconstruction** from event streams (Sec. 5).

EventNeuS outperforms multiple competing methods by at least $\approx 34\%$ on the Chamfer distance metric. The numerical evaluation also translates into substantial and consistent qualitative improvements in all tested scenarios. The source code of EventNeuS and the dataset are released[1].

2. Related Works

Event-based 3D Reconstruction. Most event-based 3D reconstruction approaches reconstruct sparse point clouds in the context of simultaneous localisation and mapping [7, 9, 27, 43]. Some approaches apply event cameras for photometric stereo capturing detailed surface normals, i.e., partial aspects of 3D object geometry [39]. Only a few methods for dense event-based 3D reconstruction of rigid [36] and non-rigid [20–22, 24, 29, 37, 44] scenes exist. EvAC3D [36] reconstructs object shapes from events, assuming those are generated by object silhouettes. It continuously carves a visual hull from apparent event contours, and, like any silhouette method, often misses fine geometric details. Next, Event-ID [3] employs SDF for implicit geometry reconstruction from multi-modal inputs, i.e., event stream and blurry RGB images. Our EventNeuS approach, in contrast, achieves detailed 3D reconstruction from events alone, without RGB supervision or prior shape templates. We adopt recent advances in neural fields for novel-view rendering and 3D reconstruction from multi-view 3D inputs [25, 35, 38]. While RGB-based methods achieve impressive 3D fidelity, they rely on standard cameras and struggle with fast motion or low-light conditions. Our EventNeuS adapts for the first time the principles of implicit surface learning to the setting with event stream supervision, enabling 3D reconstruction under challenging conditions.

Event-based Novel-View Synthesis. EventNeRF [30] pioneered NeRF learning from event streams. It provoked several follow-ups [1, 8, 10, 11, 16, 17], similarly enabling novel view rendering in the RGB space under low-light conditions from fast-moving event cameras. While all these approaches produce accurate novel views, they are not designed for accurate 3D surface reconstruction. While these approaches produce plausible novel views, they do not explicitly enforce constraints facilitating surface learning; the scene geometry remains implicit in the density field, leading to oversmoothed or missed fine details. PAEv3D [34] augments event-based NeRF with motion and geometry priors to improve reconstruction stability, yet the emphasis is still on rendering quality over exact geometry. While 3D shapes can be extracted from EventNeRF and its extensions using Marching Cubes [14], it is well known that the extracted geometry can be highly rugged and insufficiently accurate for many applications. In contrast, our method models surfaces via an SDF in addition to volumetric scene properties, which results in the recovery of more accurate 3D shape details compared to the previous state of the art.

3. Background

3.1. Event Accumulation for Learning

Event-based cameras record asynchronous per-pixel intensity changes. Each pixel independently tracks a reference value for the logarithmic brightness $L(t) = \log_\gamma I(t)$, where $I(t) \in \mathbb{R}^{W \times H}$ is the absolute intensity image at time t, W and H are the image dimensions, and γ is a gamma correction factor. When the logarithmic brightness change at a pixel reaches a predefined threshold C, the camera triggers an event $E_i = (x_i, y_i, t_i, p_i)$, where (x_i, y_i) denotes pixel coordinates, t_i is the timestamp, and $p_i \in \{-1, +1\}$ encodes the polarity. Each event corresponds to a fixed brightness change:

$$L_{x_i,y_i}(t_i) - L_{x_i,y_i}(t_{i-1}) = p_i C, \tag{1}$$

[1] https://4dqv.mpi-inf.mpg.de/EventNeuS/

where t_{i-1} is the timestamp of the previous event at the same pixel (x_i, y_i), with $p_i = +1$ for the brightness increase and $p_i = -1$ for the decrease.

Existing event accumulation strategies segment events into discrete time windows bounded by $t_i = i/N_{\text{windows}}$ for $i \in \{1, \ldots, N_{\text{windows}}\}$ [30]. Fixed windows face a trade-off: Constant short windows preserve local details but struggle to propagate high-level shape details, whereas constant long windows better capture object shape but can oversmooth high-frequency appearance details. An adaptive strategy can be used with the window start t_0 uniformly sampled from $\mathcal{U}[t - L_{\text{max}}, t)$, with L_{max} as the maximum window length. This stochastic sampling enables the joint modelling of high-frequency colour details and low-frequency lighting effects without sacrificing temporal resolution.

3.2. Wang et al.'s NeuS [35]

Recent advances in neural implicit representations have enabled high-fidelity 3D surface reconstruction from multi-view images. Among these, NeuS [35] introduces a framework that combines Signed Distance Functions (SDFs) with volume rendering to recover surfaces as the zero-level set:

$$S = \left\{ \mathbf{x} \in \mathbb{R}^3 \mid f_{\text{sdf}}(\mathbf{x}) = 0 \right\}, \tag{2}$$

where $f_{\text{sdf}} \colon \mathbb{R}^3 \to \mathbb{R}$ is a neural network predicting signed distances. Traditional volume rendering applies density-based ray marching to SDFs, but the naive policy would introduce bias because the density peak often fails to align with the true surface location. NeuS addresses this by redefining the weighting function to ensure the rendered colour correlates directly with the SDF's zero-crossing. Given a camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ with origin $\mathbf{o}$ and direction $\mathbf{d}$, NeuS computes the density $\sigma(t)$ as:

$$\alpha_i = \max\left(\frac{\Phi_s(f_{\text{sdf}}(t_i)) - \Phi_s(f_{\text{sdf}}(t_{i+1}))}{\Phi_s(f_{\text{sdf}}(t_i))}, 0 \right),$$

where $\Phi_s(x) = (1 + e^{-sx})^{-1}$ is a learnable sigmoid function scaled by parameter s, which controls the sharpness of the SDF-to-density transition. The accumulated transmittance $T(t)$ and pixel colour $\hat{C}$ are then computed as

$$T(t) = \exp\left(-\int_0^t \alpha(\tau)\, d\tau \right) \text{ and} \tag{3}$$

$$\hat{C} = \int_0^\infty T(t)\alpha(t)c(\mathbf{r}(t), \mathbf{d})\, dt, \tag{4}$$

where $c(\mathbf{r}(t), \mathbf{d})$ is the view-dependent radiance predicted by the second neural network.

While NeuS's weighting scheme is theoretically unbiased, real-world factors (e.g., sparse camera views, occlusions, or SDF estimation errors) can degrade reconstruction accuracy. While this method is likely to output high-quality reconstruction under dense and noiseless inputs, it requires additional regularisation for noisy or sparse data, as follow-ups demonstrated [12, 40]. Its probabilistic formulation ensures that surface estimates remain close to the true geometry even when biases arise from imperfect observations.

4. The EventNeuS Approach

We propose a method to reconstruct a 3D object mesh from a single event stream, eliminating the need for RGB data during training; see Fig. 2 for an overview. At the core of our approach is a neural implicit Signed Distance Function (SDF) $\varphi(\mathbf{x})$, which maps any 3D point $\mathbf{x} \in \mathbb{R}^3$ to its signed distance from the nearest surface. We adapt NeuS [35] to handle the sparse and temporally irregular nature of event data by directly linking brightness[2] changes to the geometric and photometric properties of the reconstructed 3D object. Following EventNeRF [30], we apply a Bayer filter to our predictions when computing the loss, such that only one colour channel per pixel is supervised, matching the camera model. To model view-dependent effects efficiently, we replace conventional positional encoding for camera view directions with Spherical Harmonics (SH)—marking the first use of SH in event-based 3D reconstruction to handle view-dependent effects. SH compactly represents low-frequency view variations, reducing computational overhead while improving surface detail.

Following the accumulation strategy in EventNeRF (Sec. 3.1), we convert asynchronous events into temporal windows by aggregating pixel-wise polarities. Each window encodes brightness changes as sparse frames. We assume known camera intrinsics and extrinsics for each window to ensure correct 3D-to-2D projection during training. Unlike RGB-based methods, we introduce a loss function (Sec. 4.2) that aligns temporal differences in rendered images with accumulated event data. Specifically, we minimise the mismatch between (1) brightness changes predicted by our radiance field over a time interval and (2) the observed event polarities, leveraging the event camera's inherent sensitivity to temporal gradients. Next, to synthesise images from the learned SDF and radiance field, we employ volume rendering with hierarchical sampling and frequency annealing (Sec. 4.3). We progressively increase positional encoding frequency bands during training (Sec. 4.3.2), avoiding early overfitting to noise while recovering fine details. We compute opacity $\alpha(t)$ from the SDF gradient using a logistic distribution (Sec. 4.3.4). The final pixel colour $\hat{C}$ integrates radiance values along rays, weighted by transmittance and opacity, to reconstruct surfaces consistent with event-based brightness constraints. By retaining NeuS's theoretical guarantees while operating

[2] we refer to brightness and not colour channel changes because the colours are ordered according to the Bayer filter on the pixel grid

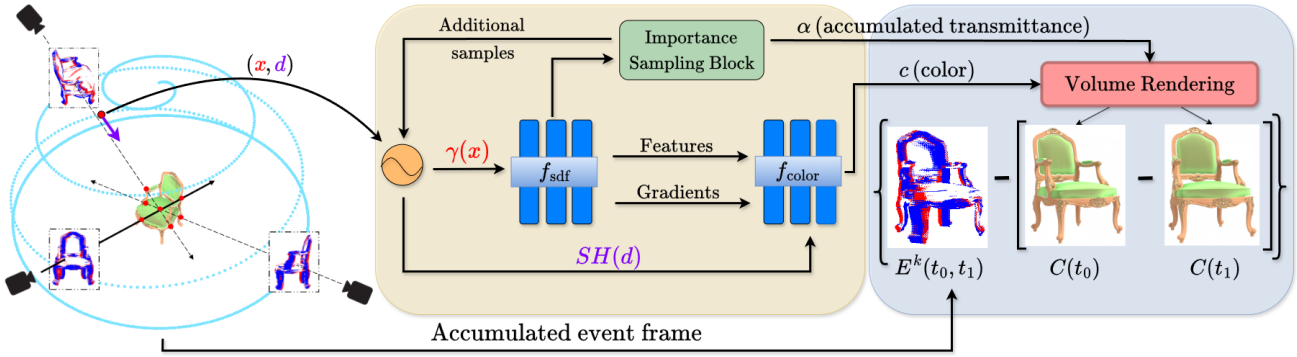

Figure 2. Overview of **EventNeuS**. We start with the trajectory that captures the object from multiple viewpoints (e.g., Seiffert's spherical spiral). We accumulate all events within a time window $[t_0, t_1]$ to form an event frame $E^k(t_0, t_1)$ (Sec. 3.1). For each event frame, we randomly choose a mini-batch of pixels and sample points along the corresponding rays. After applying positional encoding (Sec. 4.3.2) to these 3D coordinates, we feed them into the f_{sdf} network, which outputs a Signed Distance Function (SDF) and its gradients. We further refine our sampling near surfaces via importance sampling (Sec. 4.3.1). Next, we combine the resulting SDF features and gradients with view directions encoded via spherical harmonics $SH(d)$ in the f_{colour} network to predict colour (Sec. 4.3.3). Finally, we convert the SDF to a density field using Sec. 4.3.4, integrate it along each ray to obtain the *accumulated transmittance* α, and use the colour predictions to get the *rendered colour* c through volumetric rendering. We render two views (at the start and at the end of the event window) and take their difference, applying an Mean Squared Error (MSE) loss against the ground-truth accumulated event frame (Sec. 4.2). This difference enforces consistency between the rendered scene changes and the actual events recorded during $[t_0, t_1]$.

solely on event data, our approach enables robust 3D reconstruction, as validated in Sec. 5.

4.1. Event-Driven Neural Implicit Surface Learning

One of the core technical challenges of our setting is adapting NeuS for the SDF $\varphi(\mathbf{x})$ estimation from event streams. We design a loss function (Sec. 4.2) that aligns temporal differences in rendered images with accumulated event data. Let $\hat{\mathbf{C}}(t_0)$ and $\hat{\mathbf{C}}(t_1)$ denote images synthesised via volume rendering at consecutive timestamps. Minimising the discrepancy between $\hat{\mathbf{C}}(t_1) - \hat{\mathbf{C}}(t_0)$ and the observed event polarities enforces geometric consistency with the event camera's brightness change measurements (see Eq. (1)).

We model the scene geometry using an MLP approximating $\varphi(\mathbf{x})$, mapping 3D coordinates $\mathbf{x}$ to signed distances. To capture high-frequency geometric details, we apply positional encoding to $\mathbf{x}$. For appearance modelling, a separate MLP predicts view-dependent colours $c(\mathbf{x}, \mathbf{d})$ using SH encoding for view directions $\mathbf{d}$. This radiance field is conditioned on geometric features from the SDF network and surface normals $\mathbf{n} = \nabla\varphi(\mathbf{x})$. SH encoding efficiently represents low-frequency view variations, reducing computational overhead and stabilising training by avoiding overfitting to sparse event noise.

We compute opacity $\alpha(t)$ from the SDF gradient via a logistic distribution. Colours along rays are integrated using transmittance-weighted radiance values, yielding surfaces consistent with event-based brightness constraints. While NeuS assumes photometric stability across RGB frames, we directly link temporal rendering differences to event-driven brightness changes, enabling self-supervised training.

4.2. Self-Supervised Training with Event Data

Our self-supervised training objective aligns the temporal differences in the rendered images with the accumulated event data. For each camera view k, we synthesise RGB images $\hat{\mathbf{C}}^k(t_0)$ and $\hat{\mathbf{C}}^k(t_1)$ at consecutive timestamps t_0 and t_1. These predictions are aligned with the accumulated event frame $E^k(t_0, t_1) \in \mathbb{R}^{H \times W}$, which encodes brightness changes over the interval $[t_0, t_1]$. Following Event-NeRF [30], we compute the event loss $\mathcal{L}_{\text{event}}$ in logarithmic space to match the photometric response of event cameras:

$$L_{\text{event}} = \tag{5}$$
$$\text{MSE}\left(F \odot E^k(t_0, t_1),\ F \odot \left(\log\left(\hat{\mathbf{C}}^k(t_1)\right) - \log\left(\hat{\mathbf{C}}^k(t_0)\right)\right)\right),$$

where "MSE$(\cdot)$" is the mean squared error operator, "$\odot$" denotes elementwise matrix multiplication, $\hat{\mathbf{C}}^k(t) \in \mathbb{R}^{H \times W \times 3}$ is the model's predicted RGB image at time t, and F is a Bayer filter mask applied to simulate the colour sensor array. The logarithmic transform linearises intensity changes, ensuring compatibility with event camera measurements. To regularise the SDF, we apply the Eikonal loss, which enforces unit gradient norms to maintain valid SDF properties, mitigates surface artefacts and ensures smooth geometry:

$$L_{\text{eik}} = \frac{1}{N} \sum_{i=1}^{N} \left(|\nabla\varphi(x_i)|_2 - 1\right)^2, \tag{6}$$

where N is the number of sampled 3D points $\mathbf{x}_i$.

The total training loss combines both components:

$$L_{\text{total}} = L_{\text{event}} + \lambda_{\text{eik}} L_{\text{eik}}, \tag{7}$$

where $\lambda_{\text{eik}} = 0.1$ balances the regularisation strength, determined empirically across all scenes.

4.3. Volume Rendering with Hierarchical Sampling

To synthesise images from the learned SDF and radiance fields, we employ a volume rendering strategy adapted from NeuS (Sec. 3.2). Our approach extends NeuS's unbiased density transformation by introducing hierarchical importance sampling, focusing on regions with richer details along each ray. Additionally, we use a frequency annealing strategy during training to progressively improve the stability and accuracy of our model's geometry predictions.

4.3.1 Importance Sampling

We extend NeuS's unbiased density estimation with a hierarchical sampling strategy. Unlike NeuS and VolSDF [38], which use density-based PDFs derived from their respective weight functions, we concentrate samples in regions with high event activity or large SDF gradients, exploiting the event camera's inherent sensitivity to edges and texture boundaries. Initial coarse samples are drawn uniformly along each ray to estimate surface locations. Unlike NeRF's separate coarse and fine networks, we use a single network for iterative refinement, concentrating additional samples in regions with high event activity or large SDF gradients. This adaptive sampling matches NeuS's efficiency (Sec. 3.2) while focusing computational resources on regions critical for event-based reconstruction.

4.3.2 Frequency Annealing

To avoid overfitting and instability due to high-frequency noise in the early stages of training, we incorporate a frequency annealing strategy, gradually introducing high-frequency components as training progresses. This approach mitigates issues such as noisy normal maps and inaccurate geometry predictions by controlling the complexity of the frequency bands over time. The annealing coefficient $\beta_k(n)$ for the k-th frequency band at iteration n reads:

$$\beta_k(n) = \frac{1}{2}\left(1 - \cos\left(\pi \cdot \text{clamp}\left(\alpha(n) - k + N_{\text{fmin}},\ 0,\ 1\right)\right)\right), \tag{8}$$

where $\alpha(n) = (N_{f_{\text{max}}} - N_{f_{\text{min}}}) \cdot \frac{n}{N_{\text{anneal}}}$. Here, N_{fmin} (initial bands) and $N_{f_{\text{max}}}$ (final bands) control the progressive introduction of frequencies, while $N_{\text{anneal}} = 3 \cdot 10^4$ determines the annealing duration. This phased approach first stabilises low-frequency geometry before recovering fine details, reducing artefacts such as noisy surface normals.

4.3.3 Spherical Harmonics for View-Dependence

Event cameras measure temporal brightness changes influenced by view-dependent specular reflections. These event streams exhibit high sensitivity to sensor noise, which traditional view encoding methods often misinterpret as geometric detail through overfitting.

We propose spherical harmonics (SH) as the principled encoding basis for view directions, contrasting with standard positional encoding (PE). While PE operates effectively within unit-box coordinates, it becomes suboptimal for directional quantities: view directions inherently reside on the unit sphere. We address this by encoding view directions into a 16-dimensional SH basis. This SH representation is concatenated with surface normals $\nabla\varphi(\mathbf{x})$ and geometric features from the SDF network $\varphi(\mathbf{x})$.

By integrating SH-encoded directions into our radiance network f_{colour}, we resolve the ambiguity between transient artefacts and true geometric detail, enabling robust reconstruction of surfaces from sparse event streams. Implementation details can be found in App. B.

4.3.4 Opacity and Colour Integration

We next convert SDF values to densities $\sigma(t)$ using a logistic distribution peaking near the zero-level set. Opacities α_i along each ray are computed as:

$$\alpha_i = \max\left(\frac{\Phi_s(f_{\text{sdf}}(t_i)) - \Phi_s(f_{\text{sdf}}(t_{i+1}))}{\Phi_s(f_{\text{sdf}}(t_i))}, 0\right), \tag{9}$$

where Φ_s is the sigmoid function from Sec. 3.2. The final pixel colour integrates radiance values c_i weighted by transmittance T_i and opacity α_i:

$$\hat{I}_t = \sum_{i=1}^{N} T_i \alpha_i c_i, \quad \text{and} \quad T_i = \prod_{j=1}^{i-1}(1 - \alpha_j).$$

This formulation ensures that surfaces reconstructed from event data maintain photometric consistency with observed brightness changes.

4.4. Architecture and Implementation Details

Our architecture follows the NeuS [35] design, using two specialised MLPs that jointly model geometry and appearance, with modifications for event-based supervision. Once EventNeuS is trained, we query f_{sdf} and then apply Marching Cubes [15] to obtain the final mesh (and, optionally, f_{colour} for the texture). The obtained colour can be transferred to the mesh using linear interpolation of the voxel grid colours, weighted by the SDF magnitudes; see Fig. 10 in the supplementary section.

SDF Network. The geometry MLP f_{sdf} consists of eight hidden layers (256 channels each) with Softplus activations ($\beta = 100$), creating a smooth gradient landscape that stabilises SDF learning. We inject a skip connection between

the input coordinates and the fourth layer's output to enhance high-frequency detail preservation. f_{sdf} outputs both the SDF $\varphi(\mathbf{x}) \in \mathbb{R}$ and a 256-dimensional geometric feature vector $\mathbf{f} \in \mathbb{R}^{256}$ that encodes local shape properties.

Radiance Network. For view-dependent colour prediction, we employ a shallower four-layer MLP (256 channels per layer, ReLU activations) that combines four key inputs: positionally encoded 3D coordinates ($\mathbf{x}$ with eight frequency bands), surface normals $\mathbf{n} = \nabla\varphi(\mathbf{x})$, geometric features $\mathbf{f}$ from the SDF network, and view directions $\mathbf{d}$ encoded via SHs (see Sec. 4.3.3). This asymmetric architecture, the deep SDF MLP for high-capacity geometric modelling paired with a shallow radiance MLP, follows the NeuS design [35] but replaces their positional view encoding with our SH-based approach to better handle event data sparsity.

Implementation Details. Our code is based on EventNeRF [30]. We train each model for 600k iterations, which takes 45 GPU-hours on a single NVIDIA A100 GPU. At inference time, mesh extraction with Marching Cubes [15] at $300\times300\times300$ resolution takes $\approx$69.3 seconds (54.7s network queries, 8.1s surface reconstruction).

5. Experiments

5.1. Datasets

We evaluate the performance of our method on both synthetic and real-world data to validate its robustness under controlled and challenging real-world scenarios.

5.1.1 Synthetic Dataset

We generate synthetic event streams of the synthetic Chair, Mic, Hotdog, Drums, and Lego sequences [19]. We render multi-view RGB frames at a resolution of 346×260 pixels using Blender [4], capturing each object along a spherical trajectory. Following a Seiffert's spherical spiral with 8 revolutions, we extract 999 frames, resulting in an average camera speed of 2.88°/frame or 2880°/s. Event streams are then synthesised using the event-simulator [30]: The simulation pipeline converts sRGB images to logarithmic intensity space via gamma correction, applies an RGGB Bayer filter for colour-to-greyscale conversion, and triggers events when intensity differences exceed a fixed threshold of $C = 0.2$. The resulting synthetic events, camera intrinsics and extrinsics are used during training (see App. A).

5.1.2 Real Dataset

For real-world evaluation, we utilise the EventNeRF [30] dataset, including event streams captured with a DAVIS 346C colour event camera. This dataset serves as a realistic benchmark, featuring diverse structures and varying lighting conditions, to assess our model's robustness in handling practical, real-world scenarios.

5.1.3 Evaluation Metrics

We use two complementary metrics: the Mean Absolute Error (MAE) of the SDF values and the Chamfer Distance (CD). The SDF-MAE quantifies the error between the computed SDF of the ground truth and the predicted meshes, reflecting the accuracy of the volumetric reconstruction. CD assesses the geometric similarity between the two meshes by measuring the average distance between points on one mesh and their nearest neighbours on the other; see App. B for more details. Jointly, SDF-MAE and CD offer a balanced evaluation: the former emphasises volumetric consistency and the accuracy of the implicit function, while the latter directly measures the closeness of the reconstructed surfaces and explicit geometric alignment. We further use "MAE" to denote SDF-MAE for compactness.

5.2. Comparisons and Results

Our evaluation highlights significant differences in the reconstruction quality between density-based and implicit surface-based approaches. As quantified in Tab. 1, PAEv3D and EventNeRF exhibit noticeable jitter in the reconstructed surfaces (CD scores of 0.106 and 0.120, respectively), causing extracted meshes to appear noisy compared to our method's 0.07 average. This arises from insufficient constraints on 3D geometry in volume density fields (particularly in the Mic scene), where EventNeRF's CD (0.298) nearly triples our result (0.107). We emphasise that PAEv3D requires a high-resolution event stream 692×520 px for convergence, as indicated in the table footnotes, yet it still underperforms our approach across all categories.

Existing implicit surface-based methods [35], while geometrically stable (e.g., E2VID+NeuS MAE of 0.094), rely on RGB inputs and are not directly compatible with event streams. Converting events to RGB frames via existing techniques [28] introduces artefacts, as shown in the Chair scene comparisons (Fig. 3), where our method achieves 0.017 MAE versus 0.125 for the converted approach.

EventNeuS addresses both challenges by directly optimising an implicit surface representation supervised directly using event data, as evidenced by our leading scores in 9 out of 10 cases in Tab. 1. Our Eikonal loss regularises the SDF's gradient norm, enforcing smoothness while preserving geometric consistency, which is particularly effective in the Drums scene, where we achieve 0.053 CD versus 0.054/0.059 for density-based methods. This dual strategy produces meshes with notably fewer artefacts—thin structures in the Chair and Mic scenes (Fig. 3), for instance, are reconstructed with higher fidelity compared to the blurred or noisy outputs from density-based counterparts.

Table 2 additionally reports complementary novel-view synthesis metrics. See App. C.1 for a detailed discussion. Note that we were not able to include EvAC3D [36] in

	Chair		Mic		Hotdog		Drums		Lego		Avg.	
Method	Chamfer ↓	MAE ↓	Chamfer ↓	MAE ↓	Chamfer ↓	MAE ↓	Chamfer ↓	MAE ↓	Chamfer ↓	MAE ↓	Chamfer ↓	MAE ↓
E2VID [28] + NeuS [35]	0.209	0.125	0.113	0.057	0.428	0.199	0.061	0.034	0.107	0.052	0.184	0.093
EventNeRF [30]	0.061	0.030	0.298	0.138	0.102	0.043	0.054	**0.032**	0.084	0.040	0.120	0.057
PAEv3D*[34]	0.050	0.022	0.227	0.104	0.114	0.050	0.059	0.034	0.077	0.036	0.105	0.050
EventNeuS (Ours)	**0.040**	**0.017**	**0.107**	**0.052**	**0.084**	**0.037**	**0.053**	0.032	**0.067**	**0.031**	**0.070**	**0.034**

Table 1. Quantitative comparison using Chamfer Distance and MAE on NeRF synthetic scenes. Our EventNeuS achieves state-of-the-art performance across all categories, with best scores **bolded** and second-best underlined. "*": PAEv3D uses a 4x higher-resolution event stream than the rest (692×520 px), as the method does not converge otherwise.

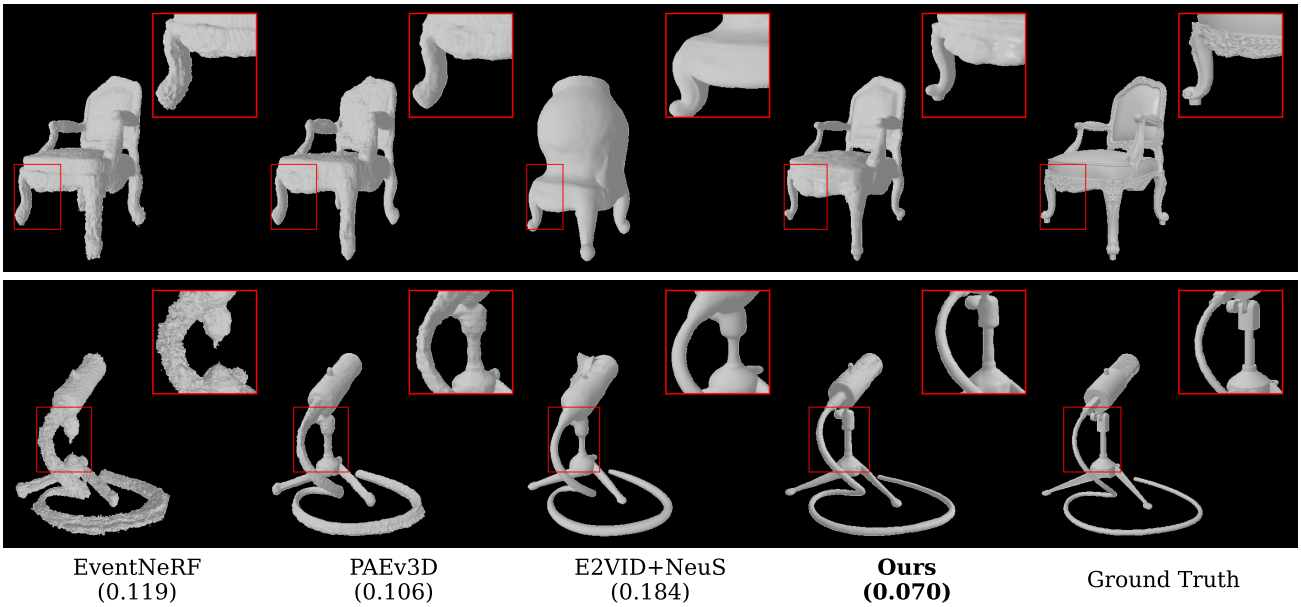

Figure 3. Qualitative comparison of our method with challenging baselines on the synthetic NeRF dataset [19] (first row: Chair; second row: Mic). Note that all results shown are **mesh reconstructions**, not volumetric renderings.

Scene	EventNeRF [30]			EventNeuS (Ours)		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Chair	**33.05**	0.98	0.06	30.94	**0.99**	**0.04**
Mic	29.61	0.98	0.02	**30.57**	0.98	0.02
Lego	23.81	0.95	0.08	**24.34**	**0.96**	**0.06**
Drums	27.83	0.97	0.05	**28.65**	**0.98**	**0.03**
Hotdog	**29.40**	0.97	0.08	28.35	**0.98**	**0.07**
Average	**28.74**	0.97	0.06	28.57	**0.98**	**0.04**

Table 2. **Complementary novel view synthesis evaluation on synthetic scenes.** While EventNeRF achieves marginally higher PSNR in two cases, EventNeuS demonstrates superior perceptual quality through improved SSIM and LPIPS scores.

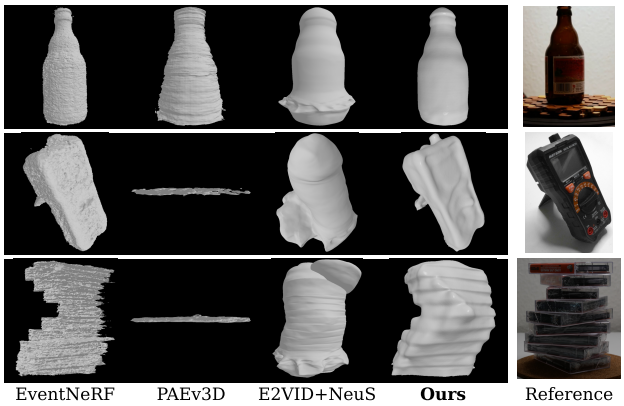

Figure 4. Qualitative comparison on the real dataset [30] with fast-rotating objects observed by an event camera. All results shown are **mesh reconstructions**, not volumetric renderings. The RGB views are provided for reference only. Best viewed with zoom.

the comparisons due to engineering reasons and incomplete source code documentation.

Fig. 4 shows that EventNeuS accurately reconstructs thin structures (e.g., multimeter and tape ridges) that baselines blur or miss, producing smoother surfaces due to robust implicit representation and effective regularisation. This not

only enables the recovery of fine geometric details but also ensures consistency across diverse scenes. These advantages make EventNeuS particularly well-suited for event-based 3D reconstruction tasks, where precise surface recovery is essential.

Method	Chamfer Distance ↓	MAE ↓
Without Negative Sampling	0.080	0.044
Without Eikonal Loss	0.077	0.044
Without SH Encoding	0.082	0.043
Without Annealing	0.079	0.042
PE Frequencies, nine steps	0.098	0.053
PE Frequencies, five steps	0.248	0.120
Full Method	**0.071**	**0.034**

Table 3. Ablation study and design choice evaluation using both CD and MAE. We report performance for several model variants after disabling negative sampling, Eikonal loss, SH encoding, and annealing strategies. Notably, the full method achieves the best performance with CD of 0.071 and MAE of 0.034, highlighting the importance of each component of our EventNeuS approach.

(a) Mic (b) Chair

Figure 5. SH encoding effectiveness on textured meshes (left: with SH, right: without SH). SH enable more detailed surfaces and better representation of view-dependent effects.

5.3. Ablation Study

We further assess the contribution of key components in EventNeuS through an ablation study, as summarised in Table 3. Removing negative sampling increases the CD and the MAE. This indicates that negative sampling is crucial for suppressing false surface detections by effectively modelling free space. Similarly, excluding the Eikonal loss leads to a slight degradation, demonstrating its importance in enforcing a smooth and well-behaved implicit surface.

The study further reveals that SH encoding is vital for capturing high-frequency appearance variations and fine geometric details. Without it (using standard positional encoding for view directions instead), the metrics deteriorate, suggesting that a more expressive view-dependent representation is crucial. Fig. 5 provides a qualitative comparison showing the effectiveness of SH encoding in the rendered textured meshes, where scenes with SH encoding exhibit superior surface detail and appearance quality. Moreover, the positional encoding (PE) frequency plays a significant role in refining the surface convergence. Experiments with different PE frequencies show that a higher frequency (nine steps) produces better accuracy compared to fewer steps. This analysis highlights the synergetic effect of our design choices and their critical role in achieving high-fidelity, event-based 3D reconstruction.

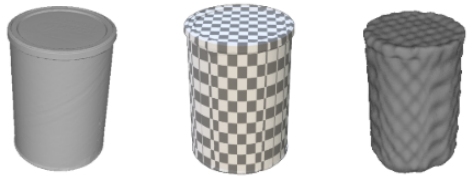

Figure 6. The left image is the ground-truth mesh, and the middle image shows the ground-truth mesh after applying chequerboard textures. The right image shows the mesh recovered with our method, with noticeable texture misinterpretation artefacts.

6. Discussion and Conclusion

As we are accumulating the events over a temporal window, we are missing some of the high-frequency temporal information in complex textures and lighting. While random window sampling solves this problem to some extent, there is still room for further research. However, incorporating more refined temporal-spatial adaptation strategies could further enhance reconstruction quality, especially in regions with high-frequency changes. We also encounter artefacts on the reconstructed mesh, such as unintended texture imprints, due to the nature of the implicit surface reconstruction method (see Fig. 6). This issue arises because the implicit surface model learns the neural surface based on texture features. Additionally, as event cameras generate events based on changes in the brightness intensity of the texture, these artefacts persist in our results as well. Our method currently does not support reconstructing large-scale scenes due to fundamental constraints, including limited network capacity for modelling expansive geometries and increased uncertainty in camera pose estimation.

Conclusion. In summary, we show that it is possible to reconstruct a 3D mesh of the observed rigid object with high accuracy, relying on the volumetric formulation solely from a monocular (possibly fast-moving) event camera. Our method achieves lower CD and SDF-MAE metrics across various objects compared to previous state-of-the-art methods, indicating closer alignment between the reconstructed and ground-truth surfaces. Qualitative assessments further confirm EventNeuS's ability to capture fine geometric details and produce smooth, accurate surface reconstructions. Future work can explore the integration of recent advances in RGB-based neural surface estimation—combined with 3D Gaussian Splatting (3DGS) [5]—into event-driven differentiable rendering frameworks, as 3DGS has been shown to be compatible with event-based supervision [24].

References

[1] Marco Cannici and Davide Scaramuzza. Mitigating motion blur in neural radiance fields with events and frames. In *CVPR*, 2024. 2

[2] Bharatesh Chakravarthi, Aayush Atul Verma, Kostas Daniilidis, Cornelia Fermuller, and Yezhou Yang. Recent event camera innovations: A survey. In *ECCV Workshops*, 2024. 1

[3] Zehao Chen, Zhan Lu, De Ma, Huajin Tang, Xudong Jiang, Qian Zheng, and Gang Pan. Event-id: Intrinsic decomposition using an event camera. In *ACM Multimedia*, 2024. 2

[4] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, 2018. 6

[5] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *CVPR*, 2024. 8

[6] Xingyi He, Jiaming Sun, Yifan Wang, Sida Peng, Qixing Huang, Hujun Bao, and Xiaowei Zhou. Detector-free structure from motion. In *CVPR*, 2024. 1

[7] Javier Hidalgo-Carrió, Guillermo Gallego, and Davide Scaramuzza. Event-aided direct sparse odometry. In *CVPR*, 2022. 2

[8] Inwoo Hwang, Junho Kim, and Young Min Kim. Ev-nerf: Event based neural radiance field. In *WACV*, 2023. 2

[9] Hanme Kim, Stefan Leutenegger, and Andrew J. Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *ECCV*, 2016. 2

[10] Simon Klenk, Lukas Koestler, Davide Scaramuzza, and Daniel Cremers. E-nerf: Neural radiance fields from a moving event camera. *IEEE Robot. Autom. Lett.*, 8(3):1587–1594, 2023. 2

[11] Wenpu Li, Pian Wan, Peng Wang, Jinghang Li, Yi Zhou, and Peidong Liu. Benerf: neural radiance fields from a single blurry image and event stream. In *ECCV*, 2024. 2

[12] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 2023. 3

[13] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *ICCV*, 2021. 1

[14] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, 1987. 2

[15] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 5, 6, 3

[16] Weng Fei Low and Gim Hee Lee. Robust e-nerf: Nerf from sparse & noisy events under non-uniform motion. In *ICCV*, 2023. 2

[17] Weng Fei Low and Gim Hee Lee. Deblur e-nerf: Nerf from motion-blurred events under high-speed or low-light conditions. In *ECCV*, 2024. 2

[18] Nico Messikommer, Carter Fang, Mathias Gehrig, and Davide Scaramuzza. Data-driven feature tracking for event cameras. In *CVPR*, 2023. 1

[19] 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. In *ECCV*, 2020. 2, 6, 7, 1

[20] Christen Millerdurai, Hiroyasu Akada, Jian Wang, Diogo Luvizon, Christian Theobalt, and Vladislav Golyanik. Eventego3d: 3d human motion capture from egocentric event streams. In *CVPR*, 2024. 2

[21] Christen Millerdurai, Diogo Luvizon, Viktor Rudnev, André Jonas, Jiayi Wang, Christian Theobalt, and Vladislav Golyanik. 3d pose estimation of two interacting hands from a monocular event camera. In *3DV*, 2024.

[22] Christen Millerdurai, Hiroyasu Akada, Jian Wang, Diogo Luvizon, Alain Pagani, Didier Stricker, Christian Theobalt, and Vladislav Golyanik. Eventego3d++: 3d human motion capture from a head-mounted event camera. *IJCV*, 2025. 2

[23] Thomas Müller. tiny-cuda-nn. https://github.com/NVlabs/tiny-cuda-nn, 2021. Version 2.0, Accessed on April 2025. 1

[24] Takuya Nakabayashi, Navami Kairanda, Hideo Saito, and Vladislav Golyanik. Ev4dgs: Novel-view rendering of non-rigid objects from monocular event streams. In *BMVC*, 2025. 2, 8

[25] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, 2021. 1, 2

[26] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 1

[27] Henri Rebecq, Timo Horstschaefer, Guillermo Gallego, and Davide Scaramuzza. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robot. Autom. Lett.*, 2(2):593–600, 2017. 2

[28] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *CVPR*, 2019. 6, 7, 2

[29] Viktor Rudnev, Vladislav Golyanik, Jiayi Wang, Hans-Peter Seidel, Franziska Mueller, Mohamed Elgharib, and Christian Theobalt. Eventhands: Real-time neural 3d hand pose estimation from an event stream. In *ICCV*, 2021. 2

[30] Viktor Rudnev, Mohamed Elgharib, Christian Theobalt, and Vladislav Golyanik. Eventnerf: Neural radiance fields from a single colour event camera. In *CVPR*, 2023. 1, 2, 3, 4, 6, 7

[31] Mohamed Sayed, John Gibson, Jamie Watson, Victor Adrian Prisacariu, Michael Firman, and Clément Godard. Simplerecon: 3d reconstruction without 3d convolutions. In *ECCV*, 2022. 1

[32] Yifei Shi, Junhua Xi, Dewen Hu, Zhiping Cai, and Kai Xu. Raymvsnet++: Learning ray-based 1d implicit fields for accurate multi-view stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2023. 1

[33] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *CVPR*, 2023. 1

[34] Jiaxu Wang, Junhao He, Ziyi Zhang, and Renjing Xu. Physical priors augmented event-based 3d reconstruction. In *ICRA*, 2024. 2, 7

[35] 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. *NeurIPS*, 2021. 1, 2, 3, 5, 6, 7

[36] Ziyun Wang, Kenneth Chaney, and Kostas Daniilidis. Evac3d: From event-based apparent contours to 3d models via continuous visual hulls. In *ECCV*, 2022. 2, 6

[37] Lan Xu, Weipeng Xu, Vladislav Golyanik, Marc Habermann, Lu Fang, and Christian Theobalt. Eventcap: Monocular 3d capture of high-speed human motions using an event camera. In *CVPR*, 2020. 2

[38] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021. 2, 5

[39] Bohan Yu, Jieji Ren, Jin Han, Feishi Wang, Jinxiu Liang, and Boxin Shi. Eventps: Real-time photometric stereo using an event camera. In *CVPR*, 2024. 2

[40] Jingyang Zhang, Yao Yao, Shiwei Li, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Critical regularizations for neural surface reconstruction in the wild. In *CVPR*, 2022. 3

[41] Yisu Zhang, Jianke Zhu, and Lixiang Lin. Multi-view stereo representation revisit: Region-aware mvsnet. In *CVPR*, 2023. 1

[42] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *ECCV*, 2018. 1

[43] Yi Zhou, Guillermo Gallego, and Shaojie Shen. Event-based stereo visual odometry. *IEEE Trans. Robot.*, 37(5):1433–1450, 2021. 2

[44] Shihao Zou, Chuan Guo, Xinxin Zuo, Sen Wang, Hu Xiaoqin, Shoushun Chen, Minglun Gong, and Li Cheng. Eventhpe: Event-based 3d human pose and shape estimation. In *ICCV*, 2021. 2