International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02197-2

Compressed Event Sensing (CES) Volumes for Event Cameras

Songnan Lin[1] · Ye Ma[1] · Jing Chen[2] · Bihan Wen[1]

Received: 3 June 2023 / Accepted: 18 July 2024

Abstract
Deep learning has made significant progress in event-driven applications. But to match standard vision networks, most approaches rely on aggregating events into grid-like representations, which obscure crucial temporal information and limit overall performance. To address this issue, we propose a novel event representation called compressed event sensing (CES) volumes. CES volumes preserve the high temporal resolution of event streams by leveraging the sparsity property of events and the principles of compressed sensing theory. They effectively capture the frequency characteristics of events in low-dimensional representations, which can be accurately decoded to raw high-dimensional event signals. In addition, our theoretical analysis show that, when integrated with a neural network, CES volumes demonstrates greater expressive power under the neural tangent kernel approximation. Through synthetic phantom validation on dense frame regression and two downstream applications involving intensity-image reconstruction and object recognition tasks, we demonstrate the superior performance of CES volumes compared to state-of-the-art event representations.

Keywords Event cameras · Data representation · Compressed sensing · Event-driven applications

1 Introduction

Event cameras are novel biologically-inspired sensors that differ from conventional cameras. Rather than capturing intensity frames at a fixed rate, event cameras respond independently and asynchronously to intensity changes for each pixel, resulting in streams of events that reflect spatiotemporal coordinates and polarity (sign) of corresponding brightness changes. Event cameras offer several promising properties, such as low power consumption, high temporal resolution (in the order of microseconds), and high dynamic range (up to 140 dB), making them a valuable alternative and complementary sensor to conventional cameras in challenging scenarios (Gallego et al., 2020).

Since event cameras produce inherently sparse and asynchronous output, it would be desirable to transform events into representations compatible with existing computer vision techniques. Spiking neural networks (SNNs) are well-suited for event data due to their minimal latency and high temporal resolution, but training them is challenging due to the lack of efficient backpropagation, making them less effective for event-driven applications (Dongsung & Terrence, 2018). In contrast, modern deep learning architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) can easily handle event data by aggregating sequences of events into grid-like representations. However, this approach has some limitations. First, grid-like representations are constructed by merging and stacking event data within a small time interval, sacrificing temporal information and leading to irreversible and lossy mapping, which limits the performance of computer vision algorithms. Second, grid-like representations do not consider the sparsity property of event data, which leads to redundant computation and may fail to highlight details. Although

Communicated by Yasuyuki Matsushita.

✉ Bihan Wen
bihan.wen@ntu.edu.sg

Songnan Lin
songnan.lin@ntu.edu.sg

Ye Ma
rudolfyema@gmail.com

Jing Chen
chen74jing29@bit.edu.cn

[1] School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Ave, Singapore 649934, Singapore

[2] School of Optics and Photonics, Beijing Institute of Technology, 5 Zhongguancun South Street, Beijing 100081, China

Published online: 31 July 2024

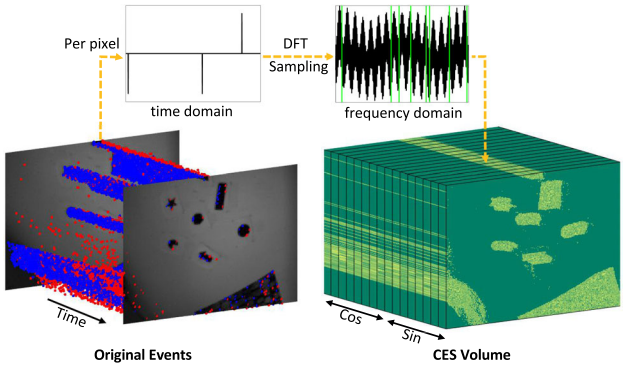

Fig. 1 CES example generated from *shapes_6dof* in the Event Camera Dataset (Mueggler et al., 2017). CES volumes are a low-loss representation that records the random frequency components of event signals

some active research areas (Yusuke et al., 2019; Jiancheng et al., 2019; Simon et al., 2022; Baldwin et al., 2022) focus on designing for sparse data, they introduce some negative effects such as computational inefficiency (Jiancheng et al., 2019; Baldwin et al., 2022) and failure to generalize to low-level tasks in the image plane (Yusuke et al., 2019; Jiancheng et al., 2019; Simon et al., 2022). (See Fig 1)

In this paper, we propose a novel event data representation called compressed event sensing (CES) volumes. Motivated by compressed sensing technique which can accurately represent a signal sparse in some domain with few samples, we utilize the sparsity property of events and compressed sensing scheme to compress event signals with a minimal loss of information. Specifically, we model the events at each pixel as a discrete-time signal. We construct the CES volumes by mapping the event signal into the frequency domain using Fourier transform and randomly selecting a subset of frequency components. The resulting volumes can be easily processed using existing computer vision algorithms while retaining the high fidelity of the original event data. The downstream applications with CES volumes reach superior performances.

The main contributions of this paper are summarized as follows:

- We propose CES volumes as an effective and efficient representation of event data. Our CES volumes hinge on the sparsity property of events and compressed sensing scheme.
- A theoretical analysis is provided showing that the proposed CES representation, integrated with a deep learning framework, demonstrates greater expressive power, using the neural tangent kernel approximation.
- We show the merits of the proposed CES, including the restricted isometry property of the sensing matrix and the high fidelity through raw event signal recovery, which are verified by empirical evidence.
- We qualitatively and quantitatively evaluate the proposed CES representation on a phantom validation of dense frame regression as well as two downstream applications: intensity-image reconstruction and object recognition.

2 Related Work

Event Representation

Owing to the asynchronous and sparse nature of event data, event-driven processing differs considerably from traditional image-driven one. Event representation is the first and critical step in the event-driven computer vision pipeline. The raw event data is transformed into an intermediary format compatible with existing techniques that implement applications such as classification, image reconstruction, and motion tracking. A comparison of event-based representations and their design choices is summarized in Table 1.

Existing algorithms designed for event cameras have traded off high temporal resolution and prediction performance. Prior approaches based on individual events gain minimal latency and high temporal resolution, such as probabilistic filters (Orchard et al., 2015; Lagorce et al., 2016) and Spiking Neural Networks (SNNs) (Zhao et al., 2014; Gehrig et al., 2020). Filter-based algorithms process events sequentially by merging the information from past events with the incoming one using a continuous-time model. However, as they hinge on handcrafted design and parameter tuning, they are less effective on more complex high-level tasks (Orchard et al., 2015; Amos et al., 2018). As for Spiking Neural Networks (SNNs) (Zhao et al., 2014; Gehrig et al., 2020), although they are more flexible by extending manual filters in a data-driven fashion, they are difficult for convergence because commercially available SNNs are still an immature technique. Moreover, these representations based on individual events are computationally intensive due to the per-event update.

To efficiently and effectively aggregate information, much attention has been paid to aggregating batches of events into grid-like representations (Zhu et al., 2019; Rebecq et al., 2019; Gehrig et al., 2019) which can be fed to traditional computer vision algorithms designed for images. Each voxel in the grid represents the accumulation of event information, e.g. count and polarity, at a particular pixel and time interval. According to the working principle of event cameras, the grid-like representations intuitively interpret scenes, such as brightness increment and edges. Moreover, they benefit from their data structure compatible with computer vision algorithms and fast inference on commodity graphics hardware. Our work is highly related to the grid-like representations above.

Recently, some efforts have been devoted to treating batches of events as a set of spatiotemporal points and learn-

Table 1 Comparison of various event representations used for event-driven applications

Representations	Dimensions	Description	Characteristics
Event Frame (Rebecq et al., 2017)	$H \times W$	Image of event polarities	Discards temporal and polarity information
Event Count Image (Ana et al., 2018; Zhu et al., 2018)	$H \times W$	Image of event coundts	Discards temporal information
Graph-based (Bi et al., 2019; Simon et al., 2022)	$U \times V$	Graph of edges and vertices	Discards temporal and spatial information
Voxel Grid (Rebecq et al., 2019)	$B \times H \times W$	Voxel grid summing event polarities	Discards event polarities
Histogram of Time Surface (HATS) (Amos et al., 2018)	$2 \times H \times W$	Histogram of average time surfaces	Discards temporal information
Time-Ordered Recent Event (TORE) (Baldwin et al., 2022)	$2 \times K \times H \times W$	4D grid of last K timestamps	Discards previous timestamps
Event Spike Tensor (EST) (Gehrig et al., 2019)	$2 \times B \times H \times W$	4D grid of convolutions	Temporally quantizes information in B bins
Compressed Event Sensing (CES)	$2 \times M \times H \times W$	4D grid of M frequency components	Compresses events with a high fidelity

H and W denote the height and width dimensions. This table is expanded from Baldwin et al. (2022) to include new representations

ing features from sparse events directly using point-based geometric processing methods, such as PointNet (Yusuke et al., 2019), transformer (Jiancheng et al., 2019), and Graph Neural Networks (GNNs) (Simon et al., 2022). These algorithms preserve the sparsity and high temporal resolution of events without redundancy, thus showing promising results on several tasks, such as object classification and detection. However, from the perspective of 3D space rather than 2D spatial one, these algorithms are not suitable for low-level tasks, such as intensity image reconstruction and optical flow estimation.

Compressed Sensing

Compressed sensing (CS) has gained tremendous interest from academic and industrial communities. By exploiting the sparsity prior, the CS scheme can reconstruct the underlying signals from much fewer samples than the Shannon-Nyquist rate (Nyquist, 1928), with the theoretical guarantee (Candès et al., 2006; Donoho, 2006). Existing works exploit various sparsity-imposing "norms", e.g. l_p or l_1 norms, for sparse coding and representation of the signals, while sensing design is guided by the mutual coherence condition or Restricted Isometry Property (RIP) (Nguyen et al., 2013). In practice, many natural signals are found to be sparsifiable, such as image (Basarab et al., 2013), geophysics data (Zhang et al., 2013), biological data (Mohtashemi et al., 2010), and communication (Bajwa et al., 2010; Eldar et al., 2012), thus CS can be applied and deployed in the corresponding applications. However, no work to date has investigated the CS of event cameras and data representation, or proposed any practical event CS schemes.

3 Methodology

In this section, we provide the problem statement of event representation, followed by the formulation of the proposed compressed event sensing (CES) method.

3.1 Event Data

Event cameras trigger an event once they detect a log-intensity change within a pixel above a preset threshold. Within a time interval, a stream of events is recorded as a set of tuples:

$$\Xi = \{\mathbf{e}_k\}_{k=1}^{K} = \{(\mathbf{u}_k, t_k, p_k)\}_{k=1}^{K}, \tag{1}$$

where $\mathbf{e}_k$ denotes the k^{th} event, $\mathbf{u}_k = (u_k, v_k)$ and t_k are the spatiotemporal coordinates, $p_k \in \{-1, 1\}$ indicates the direction (decrease or increase) of the log-intensity change.

As mentioned above, to utilize the high learning capacity of convolutional neural networks (CNNs), it is necessary to convert the event set into a grid-like representation. Ideally, this mapping should preserve the event stream's spatial structure and high temporal resolution.

3.2 Discrete-Time Event Signals

Intuitively, an event set can be viewed as several points in a four-dimensional manifold spanned by polarity and spatiotemporal coordinates. Since event cameras record events independently for each pixel, we consider an event subset $\Xi^{\mathbf{u}}$ at a specific pixel $\mathbf{u}$. Within a time interval $[0, T]$, the subset $\Xi^{\mathbf{u}}$ can be modeled as a discrete-time signal sampled with

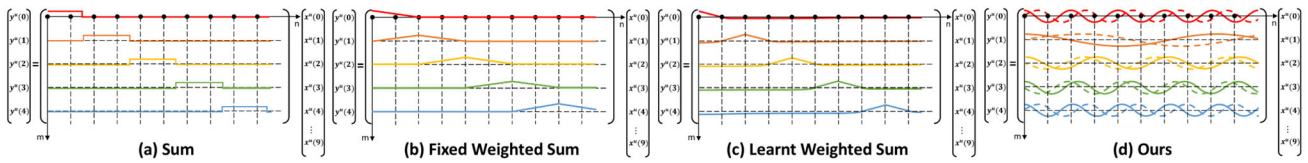

Fig. 2 Sensing matrixes Ψ of the existing grid-like event representation. Existing works are mostly based on varied sliding window schemes to compress event data, such as (a) direct sum (Songnan et al., 2020; Lin et al., 2019; Song et al., 2020; Zhu et al., 2018; Anton et al., 2019), (b) fixed weighted sum (Zhu et al., 2019; Rebecq et al., 2019), and (c) learnt weighted sum (Gehrig et al., 2019). The Ψ in (c) is recorded by generating the look-up table from the mapping learnt in the object recognition task. Our representation (d) transforms events to the frequency domain with more temporal information preserved. The real part (cosine wave) is denoted by a solid line, and the imaginary part (sine wave) by a dashed line. $M = 5$, $N = 10$ in this example

time resolution τ:

$$e^{\mathbf{u}}(t) = \sum_{n=0}^{N-1} x^{\mathbf{u}}(n\tau)\delta(t - n\tau), \tag{2}$$

where $N = \frac{T}{\tau}$ denotes the sample length, $\delta(\cdot)$ is Dirac pulse, and $x^{\mathbf{u}}(n\tau)$ is a function defined on the domain of events:

$$x^{\mathbf{u}}(n\tau) = \begin{cases} x_k & if \ \ e_k \ appear \ at \ (\mathbf{u}, n\tau), \\ 0 & if \ \ no \ events \ at \ (\mathbf{u}, n\tau), \end{cases} \tag{3}$$

where x_k denotes a measurement to event e_k. This model is used in various representations in the literature. Examples of such measurements are the event polarity (Lin et al., 2019; Songnan et al., 2020; Song et al., 2020) $x_k = p_k$, the event count (Zhu et al., 2018; Anton et al., 2019; Zhu et al., 2019; Rebecq et al., 2019) $x_k = 1$, and the normalized timestamp (Mitrokhin et al., 2018; Alonso & Murillo, 2019; Gehrig et al., 2019) $x_k = t_k/T$. We use $x_k = p_k$ in this work.

A simple approach to representing event data is to construct a 3D grid with dimensions $N \times H \times W$, where H and W denote the spatial resolution of the event camera, and N is set as the number of channels in the grid. However, due to the high temporal resolution of event cameras ($\tau \approx 1\mu s$), the resulting number of channels can be extremely large, and the sparse storage of events can lead to inefficiencies in both storage and processing. Therefore, it is desirable to compress event signals into a representation with fewer channels, while ensuring minimal loss of information.

3.3 Compressed Event Sensing

Given an event stream $\vec{x^{\mathbf{u}}} \in \mathbb{R}^N$ which is the vector format of $x^{\mathbf{u}}$, event compression can be summarized as a matrix multiplication:

$$\vec{y^{\mathbf{u}}} = \Psi^{\mathsf{T}} \vec{x^{\mathbf{u}}}, \tag{4}$$

in which $\vec{y^{\mathbf{u}}} \in \mathbb{R}^M$ is the compressed result and $\Psi = [\psi_0 \ \psi_1 \ ... \ \psi_{M-1}] \in \mathbb{R}^{N \times M}$ is a sensing matrix, $M \ll N$.

Previous works mainly adopt varied sliding window techniques to compress event data. One common approach (Songnan et al., 2020; Lin et al., 2019; Song et al., 2020; Zhu et al., 2018; Anton et al., 2019) divides an event stream into M equal-scale portions, and sums up the event measurements to form an M-channel grid, whose sensing matrix is illustrated in Fig. 2 (a). To maintain more temporal information, some methods (Zhu et al., 2019; Rebecq et al., 2019) merge events using linearly weighted summation similar to bilinear interpolation, as shown in Fig. 2 (b). Gehrig et al. (2019) further utilize a data-driven multilayer perceptron (MLP) to directly learn the best weights end-to-end for event compression. However, as shown in Fig. 2 (c), it tends to predict a simple mapping from event timestamps to weights, similar to Fig. 2 (b), and thus fails to make full use of the MLP's model capacity. Overall, according to Nyquist-Shannon sampling theorem (Nyquist, 1928), the above event compressions in the time domain inevitably discard too much temporal information, making neural networks less effective for event-driven applications.

In practice, event signals are highly sparse in the original time domain, i.e. $\|\vec{x^{\mathbf{u}}}\|_0 \ll N$. Hence, compressed sensing can be utilized to compress event signals compactly while ensuring minimal information loss with an appropriate sensing matrix Ψ. In this study, we employ random Fourier transform to generate the sensing matrix Ψ, which is widely used in natural signal processing, such as geophysical data analysis (Zhang et al., 2013), communications (Bajwa et al., 2010), and medical image processing (Basarab et al., 2013). Specifically, the Fourier transform matrix Ψ maps an input event signal $\vec{x^{\mathbf{u}}} \in \mathbb{R}^N$ into a set of compressive measurements $\vec{y^{\mathbf{u}}} \in \mathbb{R}^M$ at M random frequencies $\{f_m\}_{m=0}^{M-1}$. Each column of Ψ is defined as $\psi_m = (e^{-i2\pi f_m n\tau})^{\mathsf{T}}_{n=0,...,N-1}$, or equivalently:

$$\psi_m = [1 \quad \varpi_m \quad \varpi_m^2 \quad \varpi_m^3 \quad \cdots \quad \varpi_m^{N-1}]^{\mathsf{T}}, \tag{5}$$

where $\varpi_m = e^{-i2\pi f_m \tau}$. To avoid complex operations in following neural networks, we split the complex $\vec{y^{\mathbf{u}}}$ into its real and imaginary terms and obtain the compressed result

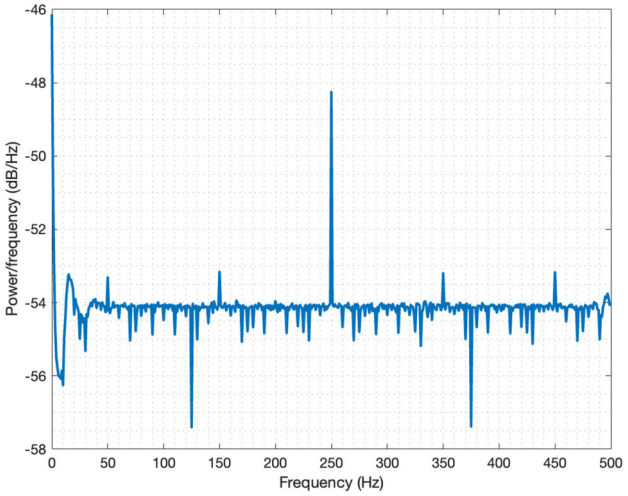

Fig. 3 Visualization on power spectral density of event data on "shapes_6dof" data (Mueggler et al., 2017). Before statistical spectral analysis, the event streams between two adjacent frames are first discretized to 1000 ($N = 1000$). Event signals show a strong structure in the frequency domain

$\vec{z^{\mathbf{u}}} \in \mathbb{R}^C$, where C denotes the channel of representation and $C = 2M$. Therefore, the c^{th} element in the representation $\vec{z_c^{\mathbf{u}}}$ is calculated by:

$$\vec{z_c^{\mathbf{u}}} = \begin{cases} \sum_{n=0}^{N-1} x^{\mathbf{u}}(n\tau) cos(2\pi f_m n\tau) & if \;\; c = 2m, \\ \sum_{n=0}^{N-1} x^{\mathbf{u}}(n\tau) sin(-2\pi f_m n\tau) & if \;\; c = 2m+1. \end{cases} \tag{6}$$

Because random Fourier transform satisfies the Restricted Isometry Property (RIP), it ensures unique and stable full signal reconstruction from fewer samples (Nguyen et al., 2013), and thus can retain much temporal information of event data.

Moreover, the frequencies can be tuned to match downstream tasks. As shown in Fig. 3, the power spectral density of event streams on "shapes_6dof" (Mueggler et al., 2017) exhibits typical characteristics of event data in the frequency domain, suggesting that the sparsely-sampled frequency components have the potential to represent events effectively. Besides, it is noteworthy that although low frequency components exhibit high energy, high frequency ones may benefit downstream tasks as well, which are not considered in the previous works (Songnan et al., 2020; Lin et al., 2019; Song et al., 2020; Zhu et al., 2018; Anton et al., 2019). Instead, our method provides greater flexibility by manually designing frequencies.

3.4 CES Implementation

After introducing our CES framework, we provide some strategies for effective and efficient implementation.

As mentioned above, the event representation with Fourier transform is tunable for downstream tasks. By choosing the frequency sampling in terms of distribution, numbers, and sampling range, it is possible to dramatically change the performance of the resulting networks. Gaussian distribution shows good flexibility by adjusting the variance σ^2, and thus we adopt Gaussian in the downstream applications:

- **Gaussian:** Randomly sample M frequencies using Gaussian distribution $f_m \sim \mathcal{N}(0, \sigma^2)$.

In principle, users can choose other distributions commonly used in the compressed sensing field (Candès et al., 2006; Nguyen et al., 2013), such as:

- **Naive FFT:** Select the first M low-order Fourier basis.
- **Uniform:** Random sample M frequencies using uniform distribution $f_m \sim \mathcal{U}[0, N]$, where N is the length of the event vectors.

Sect. 5 will provide thorough discussion on frequency sampling.

To perform the Fourier transform efficiently, we implement it using CUDA, which enables parallel computation of each pixel. Specifically, when a new event $\mathbf{e}_k = (\mathbf{u}_k, t_k, p_k)$ arrives, we assign it to the corresponding thread based on its spatial coordinate $\mathbf{u}_k$. We then update the representation $\vec{z_c^{\mathbf{u}}}$ by adding the term $\delta\vec{z_c^{\mathbf{u}}}$ as follows:

$$\delta\vec{z_c^{\mathbf{u}}} = \begin{cases} x_k cos(2\pi f_m t_k) & if \;\; c = 2m, \\ x_k sin(-2\pi f_m t_k) & if \;\; c = 2m+1, \end{cases} \tag{7}$$

where $x_k = p_k$. This update incorporates the new event into the Fourier transform representations, and enables efficient processing of large event streams.

3.5 Theoretical Analysis

The proposed CES generates measurements that retain more information from the raw event streams. However, it is unclear whether these measurements lead to more effective representations when integrated with a deep learning framework for downstream tasks. In this section, we provide a theoretical analysis showing that CES offers greater expressive power in deep learning by analyzing its Reproducing Kernel Hilbert Space using Neural Tangent Kernel approximation.

Problem Formulation Given a distinct event dataset $\mathfrak{X} = \{\vec{x_i}\}_{i=1}^{I}$, deep learning methods first represent events using a

sensing matrix Ψ and then infer using a learnable network g. Equivalently, this process is an end-to-end network p whose first layer is a fixed-weight linear transformation without activation. The output of an event input $\vec{x}$ can be formulated as

$$p(\vec{x}) = g(\Psi^{\mathsf{T}}\vec{x}). \tag{8}$$

Given a network g with a fixed architecture, the expressive power of the whole network p is fully determined by the expressive power of the chosen Ψ. Thus, the objective is to measure the function space of p, as an indicator of the expressive power of Ψ. Since p involves highly complex embedding using neural network g, we first revisit the recent neural tangent kernel (NTK) methods (Jacot et al., 2018; Tancik et al., 2020) as the tool for model simplification.

Neural Tangent Kernel (NTK)

Recent works (Jacot et al., 2018; Tancik et al., 2020; Arora et al., 2019; Bietti et al., 2019; Chen et al., 2020; Liu et al., 2023) show that, under the infinite width assumption, a neural network can be approximated as a kernel regression using NTK. Specifically, we assume that g is a fully-connected deep network with its parameters initialized from a Gaussian distribution $\mathcal{N}$. When the width of the layers in g tends to infinity, g can be approximated by kernel regression (Jacot et al., 2018). We denote its NTK function as $K_g(\cdot,\cdot)$.

Since the whole network p is essentially network g with the linear-transformed inputs $\{\Psi^T\vec{x}_i\}_{i=1}^I$, the NTK function of p can be obtained by:

$$K(\vec{x}_i,\vec{x}_j) = K_g(\Psi^T\vec{x}_i, \Psi^T\vec{x}_j). \tag{9}$$

And thus, p can also be approximated by kernel regression.

According to Bernhard et al. (2002); Seeger (2004); Saitoh et al. (2016), the reproducing kernel Hilbert space (RKHS) of a positive definite kernel encompasses a set of functions learned by kernel regression, and thus, RKHS can be used to characterise the expressive power of NTK (Chen et al., 2020). Similarly, when fixing the architecture of the network g, we propose to evaluate the expressive power of sensing matrix Ψ by comparing the RKHS of NTK corresponding to the whole network p. Formally, we introduce the following definition:

Definition 1 (Expressive Power) Given a distinct dataset $\mathfrak{X} = \{\vec{x}_i\}_{i=1}^I$, let $\mathfrak{H}_a$ and $\mathfrak{H}_b$ be the RKHS associated with the NTK of a certain network with $\Psi_a^T\vec{x}$, and $\Psi_b^T\vec{x}$ as input. When

$$\mathfrak{H}_b \subsetneq \mathfrak{H}_a, \tag{10}$$

we call the sensing matrix Ψ_a is more expressive than Ψ_b.

Comparing Reproducing Kernel Hilbert Space (RKHS)

The proposed event representation is based on compressed sensing and can satisfy the Restricted Isometry Property (RIP) with a proper sensing matrix. Therefore, for any distinct event vector pairs $\vec{x}_i, \vec{x}_j \in \mathfrak{X}$, our compressed results are also distinct, that is

$$\vec{x}_i \neq \vec{x}_j \Leftrightarrow \Psi_{ces}^T\vec{x}_i \neq \Psi_{ces}^T\vec{x}_j. \tag{11}$$

We refer to this property as the **non-degenerate property** of the sensing matrix Ψ_{ces}^T on $\mathfrak{X}$. Based on this observation, we prove the following theorem:

Theorem 1 *Given a non-zero distinct s-sparse dataset $\mathfrak{X} = \{\vec{x}_i\}_{i=1}^I$, let $\mathfrak{H}_a$ and $\mathfrak{H}_b$ be the RKHS associated with the NTK of same-architecture fully-connected network with $\Psi_a^T\vec{x}$, and $\Psi_b^T\vec{x}$ as input, where Ψ_a holds the non-degenerate property while Ψ_b does not, the following subset inclusion relation hold:*

$$\mathfrak{H}_b \subsetneq \mathfrak{H}_a. \tag{12}$$

Please refer to the appendix 7 for detailed proof.

The proposed CES has the non-degenerate property, while the existing grid-like representations do not. According to **Theorem** 1, under the NTK assumption, the neural networks with CES can reach a larger RKHS compared with other representations. And thus, based on **Definition** 1, we can conclude that our CES exhibits greater expressive power.

4 Fidelity of CES Volumes

The existing representations of event data often suffer from an irreversible loss of temporal information. In contrast, our proposed compressed event sensing (CES) approach can encode the full temporal information of the event data with minimal loss. This section discuss on the quality of the sensing matrices of existing event representations and further evaluate our high fidelity by recovering raw event signals from our proposed CES volumes.

4.1 Quality of Sensing Matrix

According to the theory of compressed sensing (Simon & Holger, 2013), restricted isometry property (RIP) is a fine measure of the quality of a measurement matrix. The sth restricted isometry constant $\delta_s = \delta_s(\Psi^T)$ of the sensing matrix Ψ^T is the smallest $\delta \geq 0$ such that

$$(1-\delta)\|\vec{x^{\mathbf{u}}}\|_2^2 \leq \|\Psi^T\vec{x^{\mathbf{u}}}\|_2^2 \leq (1+\delta)\|\vec{x^{\mathbf{u}}}\|_2^2, \tag{13}$$

for all s-sparse event vector $\vec{x^{\mathbf{u}}} \in \mathbb{R}^N$. Equivalently, it is given by:

$$\delta_s = max_{S\subseteq[N],card(S)\leq s}\|\Psi_S\Psi_S^T - I_N\|_{2\to 2}. \tag{14}$$

Table 2 Comparison on the quality of the sensing matrix Ψ^T with different representations in terms of restricted isometry constant δ_s for sparse vectors with $s = 2$ sparse and $N = 1000$ length

Channel number	4	8	16	32	64
Voxel Grid (Rebecq et al., 2019)	1.000	1.000	1.000	1.000	1.000
EST (Gehrig et al., 2019)	1.000	1.000	1.000	1.000	1.000
Ours	0.991	0.834	0.607	0.452	0.304

The lower δ_s, the higher quality of the sensing matrix. The sensing matrix of the proposed CES obtains lower δ_s under different channel numbers, demonstrating its effectiveness on event representation

If δ_s is small for reasonably large s, the sensing matrix satisfies the RIP and ensures a stable and unique sparse recovery.

Table 2 compares the δ_s for N-length s-sparse event vectors ($N = 1000, s = 2$) among the sensing matrices Ψ^T of different representations. Compared with existing representations, our method achieves lowest restricted isometry constants for all numbers of frequencies M. And thus, it exhibits high probability of success to recover sparse signal via algorithms.

Moreover, we observed that the quality of our sensing matrix improves as the number of sampled frequencies increases. According to Candès et al. (2006), the number of sampled frequencies required for accurate recovery via $l1$ minimization should satisfy: $M \geq C_M \cdot logN \cdot \|\vec{x^{\mathbf{u}}}\|_0$, for some constant $C_M > 0$. Therefore, it would be necessary to design a suitable channel number of representations based on the sparsity of the event signal $s = \|\vec{x^{\mathbf{u}}}\|_0$ and the desired temporal resolution N.

4.2 Raw Event Signal Recovery

We conduct an experiment on the "shapes_6dof" dataset (Mueggler et al., 2017) captured by DAVIS240C. Each event stream between two adjacent frames with a duration of approximately 40 ms is compressed by CES volume and then reconstructed by a conventional sparse recovery algorithm. However, due to the high temporal resolution of event cameras ($\tau \approx 1\mu s$), the vector format of event streams is very lengthy ($N \approx 40,000$), which poses computational challenges for the conventional sparse recovery algorithms. To address this problem, we conduct a rough evaluation experiment by shortening event signal N as an alternative. Specifically, we normalize the timestamps of event data to a range of [0, 1] and discretize them into N=1000 intervals. We randomly sample $M = 4, 8, 16, 32, 64$ frequencies $\{f_m\}_{m=0}^{M-1}$ from a uniform distribution over the interval $[0, N]$, and calculate corresponding frequency components at each pixel to form a grid-like representation. Then, we feed the representations into the iteratively re-weighted $l1$ minimization (Candes et al., 2008). Through examining the quality of the reconstructed signals, we can evaluate the fidelity of our representation.

We evaluate the average PSNR between the original events and the reconstructed ones as shown in Table 3. We also provide qualitative results in Fig. 4, where the decompressed event data are highlighted in panels (b), (c), and (d) using a threshold of 0.5 for the absolute values of signal intensities.

Table 3 Raw event signal recovery from the proposed CES volumes with different numbers of frequencies M on "shapes_6dof" dataset (Mueggler et al., 2017)

Frequency number	4	8	16	32	64
PSNR ↑	46.65	49.37	59.26	212.04	221.87

Compared with existing representations suffering from irreversible losses, CES volumes preserve more temporal information and enable the raw event signal recovery

The results demonstrate that our proposed representation, which relies on the compressed sensing scheme, excels in reconstructing realistic events with remarkable fidelity. Furthermore, we observed that the performance of our method improves as the number of sampled frequencies increases, which is consistent with the observation in Table 2. When M is small, such as in Fig. 4 (b), the insufficient number of frequencies causes missing and misaligned signals during sparse recovery. However, when a sufficient number of frequencies are sampled, as in Fig. 4 (c) and (d), the restored events closely resemble the original ones.

However, in practice, downstream applications may not rely on temporal information as heavily as the raw event signal recovery does. The detailed discussion on frequency numbers will be given in Sect. 5.

5 Experimental Results

In this section, we first investigate the advantage of CES volumes on preserving much temporal information by a challenging case: dense frame regression test on a synthetic phantom dataset. Then, we demonstrate the effectiveness of our proposed CES volume on two event-driven applications: intensity-image reconstruction and object classification. Finally, we compare the running time of the existing representations.

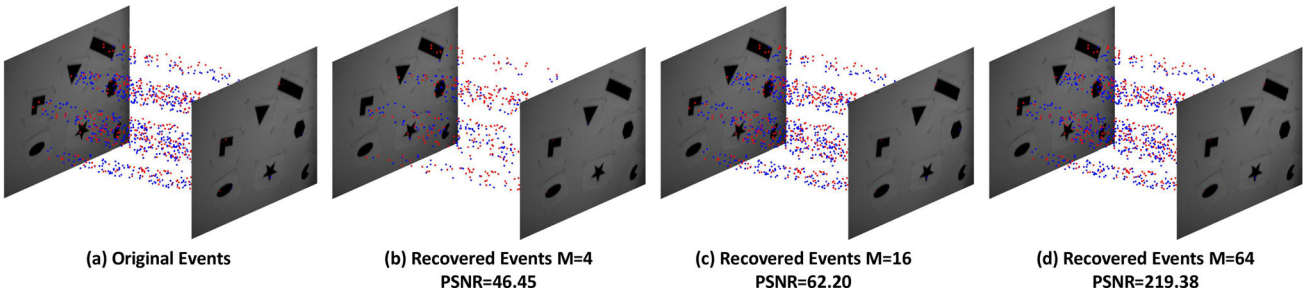

Fig. 4 Visualization on the reversibility of the proposed CES volumes on "shapes_6dof" data (Mueggler et al., 2017). Different from existing event representations suffering from irreversible losses, the proposed CES can be used to accurately recover the raw event signals. (a) denotes an original event signal between two adjacent frames. (b)(c)(d) show the recovered event signals by feeding our representations with the frequency number $M = 4,\ 16,\ 64$ into a conventional sparse recovery algorithm (Candes et al., 2008) and illustrating the points where the absolute values of signal intensities are above a threshold 0.5. The recovered results ensemble the original event signal and reach PSNR of 46.45, 62.20 and 219.38, respectively. The experiment shows the reversible recovery ability of the proposed CES volumes and demonstrates its powerful representational capability

5.1 Dense Frame Regression Test

To better understand the proposed method's advantage of keeping high temporal resolution, we regress dense frames from a starting frame and the subsequent events on a simulated phantom dataset. Preserving more temporal information should result in better reconstruction of dense frames.

Given a starting intensity $z_0^{\mathbf{u}} \in \mathbb{R}$ at a pixel $\mathbf{u}$ and the subsequent event stream $\vec{x^{\mathbf{u}}} \in \mathbb{R}^N$, dense frame regression aims to output corresponding frames $[z_1^{\mathbf{u}}, z_2^{\mathbf{u}}, ..., z_{Z-1}^{\mathbf{u}}] \in \mathbb{R}^{Z-1}$. This procedure can be formulated as

$$[z_1^{\mathbf{u}}, z_2^{\mathbf{u}}, ..., z_{Z-1}^{\mathbf{u}}] = z_0^{\mathbf{u}} + f(\vec{y^{\mathbf{u}}}; \theta) = z_0^{\mathbf{u}} + f(\Psi^{\mathsf{T}} \vec{x^{\mathbf{u}}}; \theta) \tag{15}$$

in which $f(\cdot)$ is a fully-connected networks (also called multilayer perceptrons or MLPs) with weights θ, $\Psi \in \mathbb{R}^{N \times M}$ is a sensing matrix and $\vec{y^{\mathbf{u}}} \in \mathbb{R}^M$ is the compressed representation. The pipeline of dense frame regression is illustrated in Fig. 5 (a).

Specifically, we compare the proposed CES volumes with the state-of-the-art representations, including fixed weighted sum-based Voxel Grid (Zhu et al., 2019; Rebecq et al., 2019) and learnt weighted sum-based Event Spike Tensor (EST) (Gehrig et al., 2019). The timestamps of events are normalized to a range of [0, 1]. As for our CES volumes, we sample the frequencies from a Gaussian distribution $f_m \sim \mathcal{N}(0, \sigma^2)$ similar to (Tancik et al., 2020), and form the volumes with corresponding frequency components. For a fair comparison, we set the same number of channels of each representation $C = 16$.

The network $f(\cdot)$ contains 8 fully-connected (FC) layers. The channel number of the last layer is set as $Z - 1$, where $Z = 50$ in this experiment. Other layers are designed as 256 channels and followed by GELU activations (Hendrycks et al., 2016). We implement the network using PyTorch (Paszke et al., 2017) and use ADAM (Diederi amb Jimmy, 2014) with a learning rate of 0.001. The networks are trained with the mean squared error (MSE) loss for 50 epochs (937,500 iterations) with a batch size of 16.

Datasets To obtain the phantom dataset, we first generate several video sequences by rendering a brighter rectangle with the size of 50×100, on a darker background with the size of 100×100. The rectangle starts at the center of the background, and moves horizontally for 3 key times with various and random step lengths, otherwise keeps stationary. Each sample contains Z frames. Figure 5 (b)-(e) show an example in which the rectangle moves at 21st, 47th, and 49th frames. After obtaining video sequences, we feed them into an event simulator, Vid2E (Gehrig et al., 2020), to generate an event stream. The dataset contains 300,000 video samples and corresponding events. We select 225,000 pairs for training and 75,000 pairs for testing.

Results and Discussion Figure 6 provides the convergence of existing event representations. Existing representations converge rapidly but show limited power in frame regression. Based on compressed sensing theory, the proposed CES volumes capture frequency characteristics of events and encode much temporal information, and thus have potential to distinguish frames in dense timestamps. It shows good learning capacity of dense frame regression.

Table 4 compares the mean squared error (MSE) and structural similarity (SSIM) (Zhou et al., 2004) of regression results. Voxel Grid (Rebecq et al., 2019) and EST (Gehrig et al., 2019) utilize a sliding window approach, which fuses a large number of events within each window, leading to blurred temporal information and limited ability to recover high time resolution. In contrast, the proposed method utilizes compressed sensing scheme to preserve more temporal information, thereby overcoming the "temporal

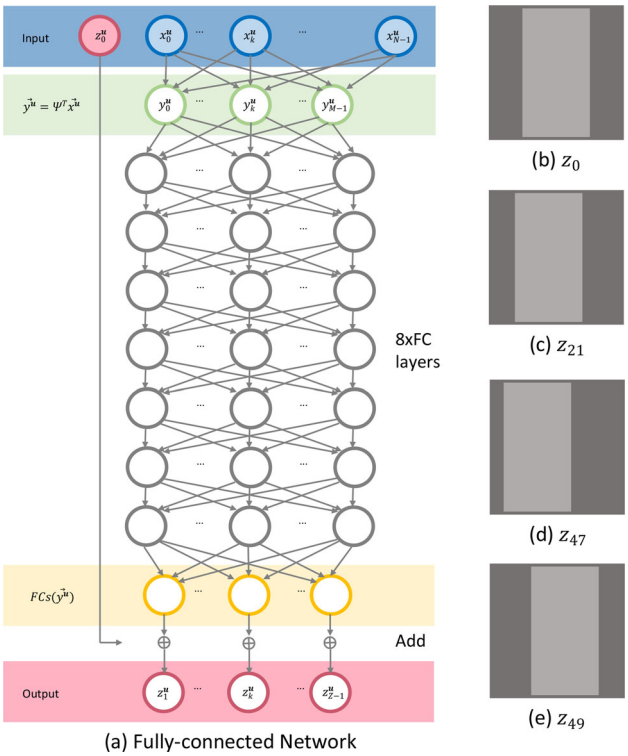

Fig. 5 Pipeline and dataset synthesis of dense frame regression test. Given a starting frame $z_0^{\mathbf{u}}$ at a pixel $\mathbf{u}$ and its subsequent event stream $\vec{x^{\mathbf{u}}}$, dense frame regression aims to predict corresponding subsequent frames $[z_1^{\mathbf{u}}, z_2^{\mathbf{u}}, ..., z_{Z-1}^{\mathbf{u}}]$. The event data $\vec{x^{\mathbf{u}}}$ is first represented as $\vec{y^{\mathbf{u}}}$ using different sensing matrices Ψ^T, and then fed into a fully-connected network. As for the synthetic phantom dataset, we generate video sequences by rendering a brighter rectangle moving on a darker background, and simulating event data using Vid2E (Gehrig et al., 2020). Please see manuscripts for more details

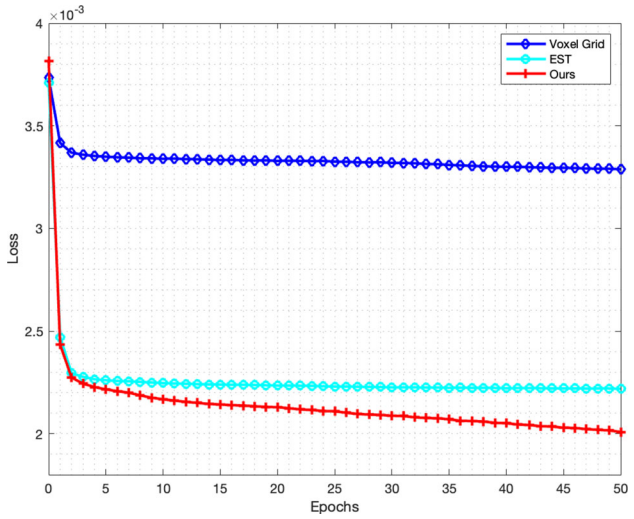

Fig. 6 Convergence of dense frame regression. Compared with existing representations, the proposed compressed sensing-based method preserves more temporal information of events, showing good learning ability on regressing and distinguishing dense frames

bias" observed when training networks with sliding window. It allows for the distinction of dense timestamps and results in high regression performance.

Actually, it comes as no surprise to learn that our frequency domain representation performs favorably. Previous work (Tancik et al., 2020) has demonstrated on a variety of regression tasks relevant to the computer vision and graphics communities that Fourier feature mapping can overcome the spectral bias of coordinate-based MLPs towards low frequencies by allowing them to learn much higher frequencies.

5.2 Intensity-Image Reconstruction

Intensity-image reconstruction aims to generate high-quality video frames from sparse event data, allowing for the use of off-the-shelf computer vision techniques designed for conventional cameras. Ideally, benefiting from the pleasing properties of events, the reconstructed intensity images can exhibit a significantly larger dynamic range, shaper edges, and reduced motion blur compared to conventional cameras.

However, there are several challenges associated with reconstructing high-quality images from event data. First, events are unevenly distributed in both space and time, with the number of events being highly dependent on the movement in the scene. In the presence of large movements, the captured events can be massive, posing difficulties in event representation, as essential information may be lost, resulting in blurry reconstructed images. On the other hand, small movements can lead to sparser events, requiring event representation methods to effectively highlight the relevant details without sacrificing important information.

Furthermore, the imbalance in the amounts of events introduced by different object motions adds another layer of complexity to the design of data representation methods. Finding a suitable event representation approach that

Table 4 Dense frame regression performance on the synthetic phantom dataset in terms of mean squared error (MSE) and structural similarity (SSIM) (Zhou et al., 2004)

Methods		Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	Ours
MSE	↓	0.0032	0.0022	0.0020
SSIM	↑	0.8309	0.8879	0.8952

The proposed method, based on compressed sensing theory, preserves more temporal information of events, allowing it to distinguish dense frames and achieve high regression performance

can capture both large and small movements while preserving the essential details poses a significant challenge in intensity-image reconstruction from event data. Addressing these challenges is crucial to achieving high-quality image reconstruction from event data and unlocking the full potential of event cameras in computer vision applications.

In this section, we conduct an evaluation on a state-of-the-art high-pass filter-based non-deep method (Cedric et al., 2018) as well as a deep learning-based method E2VID (Rebecq et al., 2019) integrating various representations, including fixed weighted sum-based Voxel Grid (Zhu et al., 2019; Rebecq et al., 2019), learnt weighted sum-based Event Spike Tensor (EST) (Gehrig et al., 2019), TORE (Baldwin et al., 2022) and our compressed event sensing method. To ensure a fair comparison, we set the number of channels for each representation to be the same, with $C = 8$. These representations are then fed into the same base network, E2VID (Rebecq et al., 2019), for further processing.

We implemented the networks using PyTorch (Paszke et al., 2017) and adopted ADAM (Diederi amb Jimmy, 2014) as the optimization algorithm, with a learning rate of 0.0001. The networks were trained for 20 epochs, corresponding to 56,540 iterations, with a batch size of 2.

Datasets In our experiments, we utilize the slow subset of the Blur-DVS dataset (Jiang et al., 2020) for both training and validation. The slow Blur-DVS dataset was captured using a DAVIS240C event camera with slow and stable camera movement, capturing relatively static scenes. As a result, this dataset provides sharp intensity frames as ground truths and corresponding event streams as inputs, which are suitable for training our model. The dataset is split into 11,308 pairs for training and 3,700 pairs for validation.

To further assess the generalization ability of the representations, we also conducted experiments on the Event Camera Dataset (Mueggler et al., 2017). This dataset presents a different set of scenes and events compared to the Blur-DVS dataset.

Results and Discussion We report mean squared error (MSE), structural similarity (SSIM) (Zhou et al., 2004), and the learned perceptual image patch similarity (LPIPS) (Richard et al., 2018) in Table 5. On Blur-DVS dataset, the proposed representation method demonstrates superior performance with an 8% decrease in MSE, a 3% increase in SSIM, and a 2% decrease in LPIPS. Our method exhibits better generalization capability across almost all scenes, as indicated by improvements in MSE (1% decrease), SSIM (4% increase) and LPIPS (5% decrease).

We provide visual examples from the validation and testing sets in Figs. 7 and 8, respectively. Voxel Grid (Rebecq et al., 2019), which relies on a fixed weighted sum, sacrifices temporal information and is less effective in handling significant camera movement, resulting in noticeable ringing artifacts. Event Spike Tensor (EST) (Gehrig et al., 2019), which uses learnable weights for event data embedding, exhibits limited learning ability, and the learnable weights are similar to the fixed weights in Voxel Grid (see Fig. 2), leading to blurry intensity images. In contrast, our proposed representation method leverages the frequency characteristics and sparsity of events to encode data using a compressed sensing scheme, resulting in reconstructed images with finer details and fewer visual artifacts.

Effect of Frequency Number

According to the compressed sensing theory, our proposed CES volumes can retain extensive temporal information when a sufficient number of frequencies are sampled. Nevertheless, in event-driven applications, preserving accurate temporal resolution with numerous frequencies may not always be unnecessary. To further explore this, we carry out an experiment where we vary the number of channels in the representations representations, setting them to $C = 8, 16, 32, 64$, and evaluate their performance on an intensity-image reconstruction task, as shown in Fig. 9. We compare our method with Voxel Grid (Rebecq et al., 2019) and Event Spike Tensor (EST) (Gehrig et al., 2019).

We observe that the accuracy of all methods generally increase with the number of channels. However, our method exhibits greater robustness to the number of channels and consistently achieves superior performance across all channel levels. These findings suggest that intensity-image reconstruction may be less sensitive to temporal information compared to event signal reconstruction, and satisfactory performance can be achieved with a limited number of frequencies using in our proposed method. This makes our method valuable in scenarios where computation or storage constraint exists.

Effect of Frequency Sampling Range

The tunable nature of the proposed representation allows for manipulation of frequency settings to optimize downstream network performance. To investigate the effects of frequency sampling range on the intensity-image reconstruction task, we conduct an experiment where the frequency number is set as 4, 8, 16, 32, 64 and the frequencies are sampled from a Gaussian distribution with variance σ^2 ranging from 1 to 25.

Figure 10 illustrates the interaction on intensity-image reconstruction in terms of (a) MSE, (b) SSIM, and (c) LPIPS. We note that given a fixed channel number of event representations, the image reconstruction accuracy initially improves and then declines with an increase in sampling variance. Moreover, as the channel number increases, the inflection point occurs later and the reconstruction network obtains higher accuracy.

The observation may stem from 2 reasons. 1) Low-frequency components reflect the overall trend of event streams, which is crucial and necessary in the intensity-image reconstruction application. Gaussian distribution with

Table 5 Intensity-image reconstruction performance compared with existing grid-like representations and state-of-the-art reconstruction methods on the Blur-DVS (Jiang et al., 2020) and Event Camera Dataset (Mueggler et al., 2017) in terms of mean squared error (MSE), structural similarity (SSIM) (Zhou et al., 2004), and the learned perceptual image patch similarity (LPIPS) (Richard et al., 2018)

	MSE ↓					SSIM ↑					LPIPS ↓				
	HF (Cedric et al., 2018)	Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	TORE (Baldwin et al., 2022)	Ours	HF (Cedric et al., 2018)	Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	TORE (Baldwin et al., 2022)	Ours	HF (Cedric et al., 2018)	Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	TORE (Baldwin et al., 2022)	Ours
Blur-DVS	0.102	0.039	0.037	0.061	**0.034**	0.426	0.705	0.724	0.368	**0.743**	0.648	0.480	0.465	0.653	**0.455**
box_6dof	0.157	0.061	0.062	0.044	**0.057**	0.350	0.620	0.645	0.393	**0.674**	0.675	0.491	0.475	0.627	**0.453**
calibration	0.189	**0.064**	0.067	0.065	0.068	0.382	0.659	0.636	0.311	**0.673**	0.758	0.489	0.515	0.691	**0.483**
dynamic_6dof	0.252	0.128	0.136	**0.096**	0.130	0.219	0.334	**0.369**	0.195	0.354	0.674	0.528	**0.497**	0.598	0.503
office_zigzag	0.145	0.061	0.070	0.060	**0.060**	0.374	0.586	0.541	0.242	**0.596**	0.716	0.542	0.532	0.665	**0.499**
poster_6dof	0.219	0.107	0.112	**0.074**	0.105	0.229	0.468	0.513	0.322	**0.526**	0.619	0.420	0.391	0.569	**0.369**
shapes_6dof	0.152	0.065	0.054	**0.024**	0.063	0.553	**0.565**	0.501	0.531	0.537	0.637	**0.490**	0.546	0.647	0.503
sliding_depth	0.164	0.077	0.066	**0.049**	0.071	0.369	0.482	**0.566**	0.340	0.554	0.730	0.599	0.525	0.632	**0.519**
mean	0.189	0.083	0.084	**0.060**	0.082	0.351	0.535	0.536	0.344	**0.558**	0.678	0.490	0.499	0.630	**0.466**

For a fair comparison, the grid-like representations have the same channels $C = 8$ and are fed into the same E2VID (Rebecq et al., 2019) network

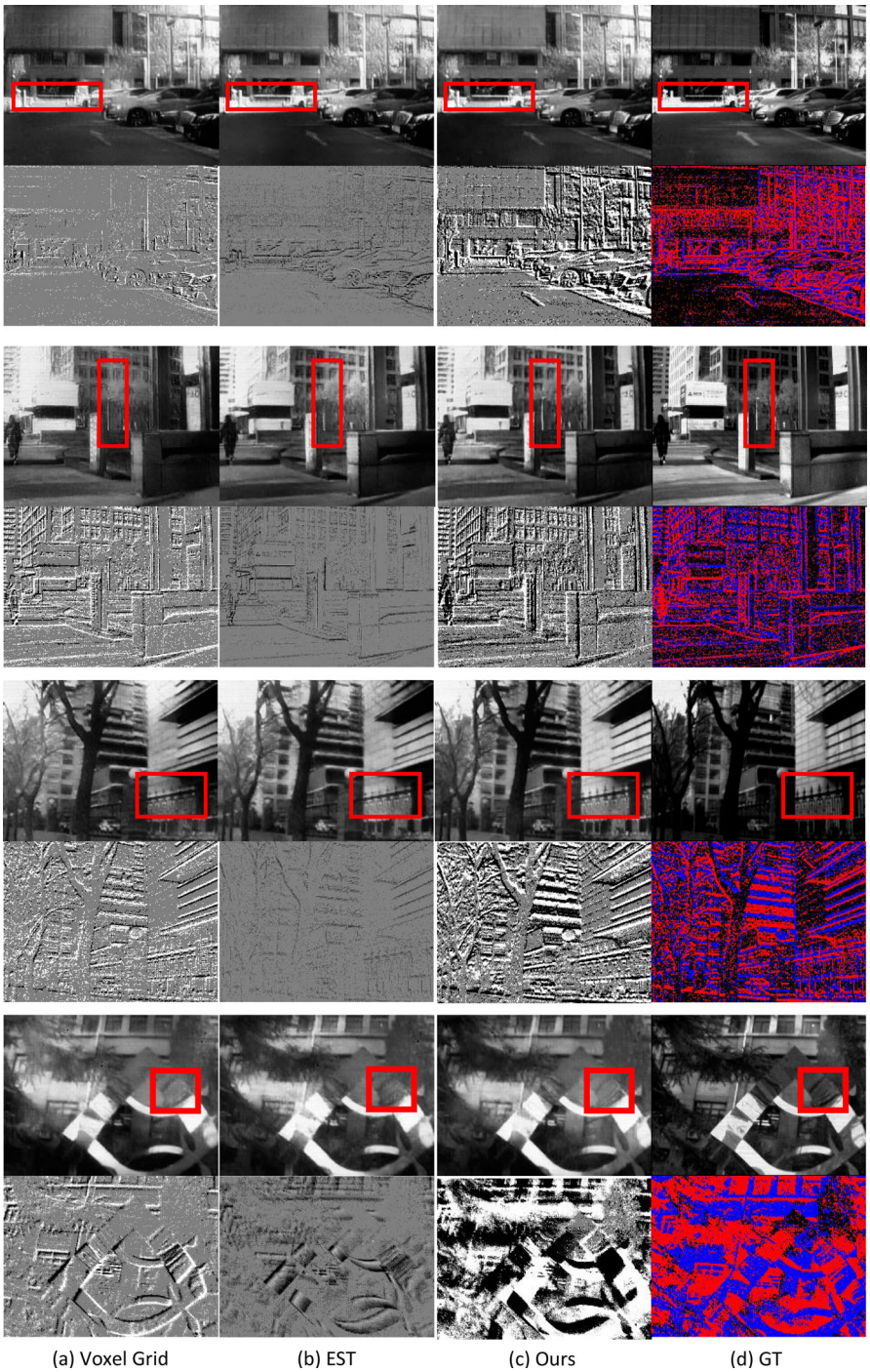

Fig. 7 Visual comparisons on intensity-image reconstruction on Blur-DVS (Jiang et al., 2020). The reconstructed images from Voxel Grid (Rebecq et al., 2019), EST (Gehrig et al., 2019), and our representation as well as ground truth are shown in the odd rows, while the center channel of each representation and input event data are shown in the even rows. The network designed with our representation generates more details with fewer artifacts

small variances guarantees the existence of low frequency components, while high variances does not. Therefore, the accuracy decreases with the increase of sampling variance. 2) The richness of the sampled frequencies is crucial. When the channel is large but the variance is small, the generated frequencies becomes redundant, leading to limited expressive capacity in the representation and a loss of detailed information. Therefore, the accuracy improves initially with an increase in sampling variance. It also explains why the inflection point in 2D plots occurs later with an increase in channel number.

Effect of Frequency Sampling Distribution

We have discussed the effect of frequency sampling range using Gaussian distribution above, and in this section, we further investigate the impact of different frequency sampling distributions. We conduct an experiment on uniform distribution as well as “naive FFT" which selects the first a few low-order Fourier basis instead of random set of basis.

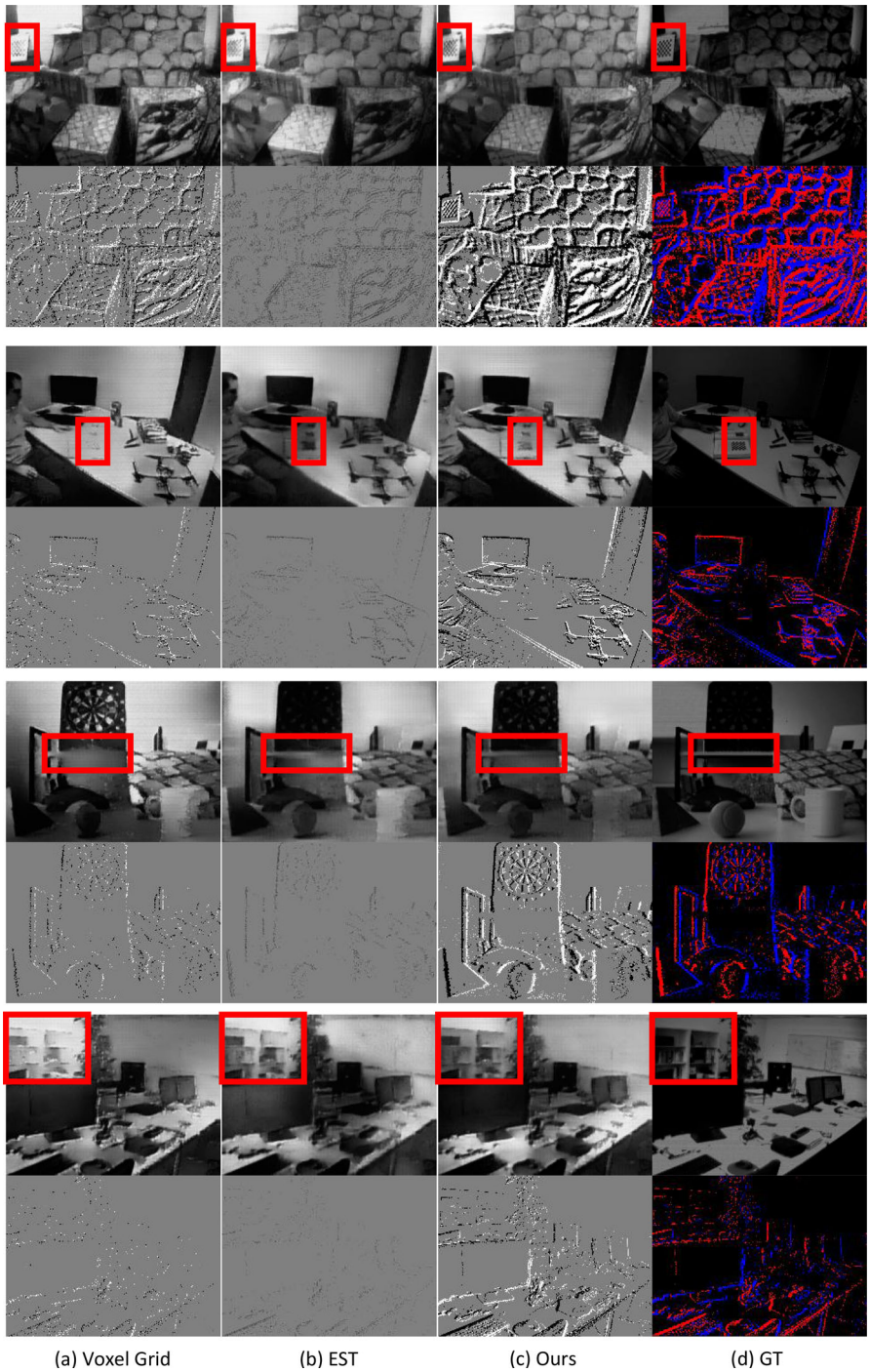

Fig. 8 Visual comparisons on intensity-image reconstruction on Event Camera Dataset (Mueggler et al., 2017). The reconstructed images from Voxel Grid (Rebecq et al., 2019), EST (Gehrig et al., 2019), and our representation as well as ground truth are shown in the odd rows, while the center channel of each representation and input event data are shown in the even rows. The network designed with our representation generates more details with fewer artifacts

Figure 11 illustrates the comparison results. The “naive FFT" ensures the existence of low-frequency components and the richness of frequencies similar to Gaussian distribution, and thus obtain comparable reconstruction results with Gaussian distribution with suitable variances. However, given a limited of frequency number, uniform distribution cannot ensure the inclusion of enough low-frequency components and thus presents poor reconstruction performance.

Actually, determining the optimal sampling distribution is challenging and there is no theoretical justification or guarantees. Different applications may have varying requirements of frequencies on event representation. Moreover, the complexity and intrinsic characteristics of event stream may influence the optimal frequency sampling as well.

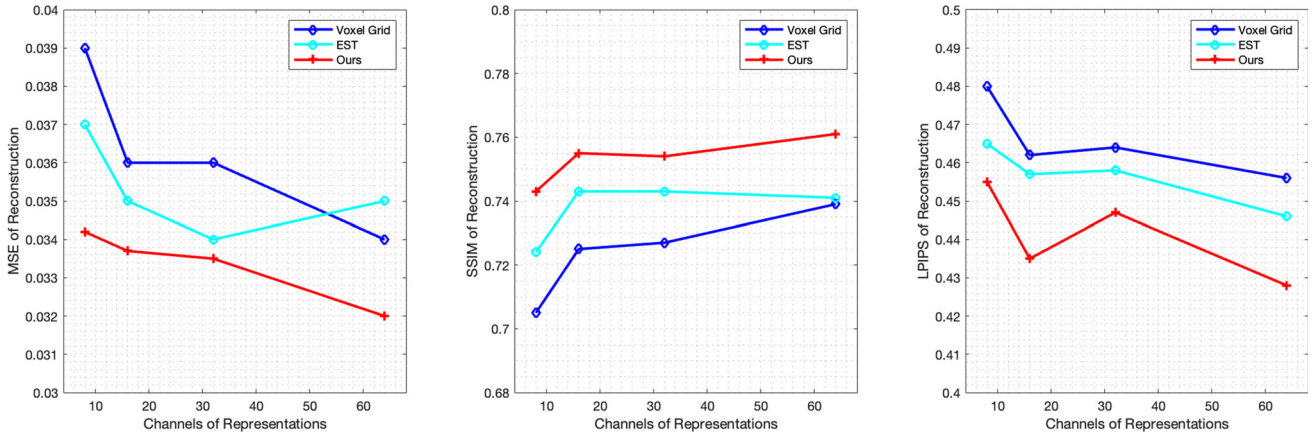

Fig. 9 Effects of channel numbers of representations on intensity-image reconstruction on Blur-DVS (Jiang et al., 2020). The proposed method achieves the highest performance under all numbers of channels. It can perform favorably with few channels, resulting in efficient representation

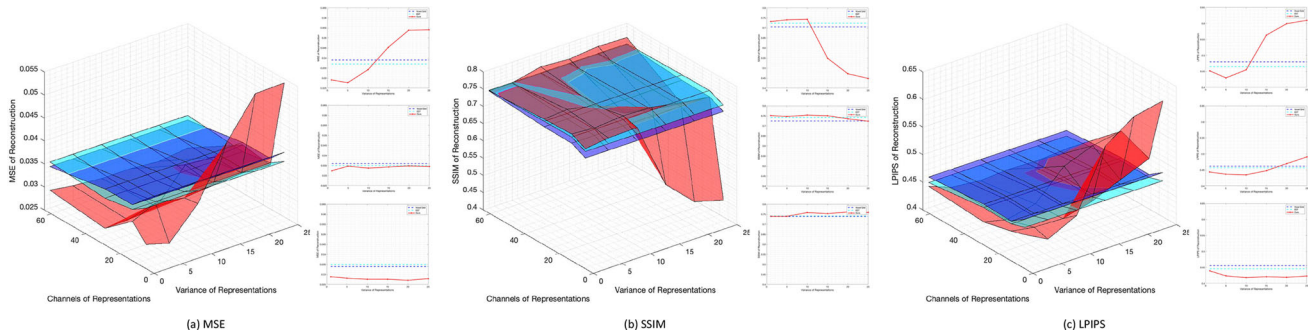

Fig. 10 Comprehensive analysis on the interaction between channel numbers and sampling variances on intensity-image reconstruction on Blur-DVS (Jiang et al., 2020). Voxel Grid (Rebecq et al., 2019), EST (Gehrig et al., 2019), and the proposed method are highlighted as blue, cyan, and red, specifically. The 3D surface plots illustrate the interaction on (a) MSE, (b) SSIM, and (c) LPIPS. The channel numbers are set as 8, 16, 32, 64; while frequencies are sampled from a Gaussian distribution with variance σ^2 ranging from 1 to 25. The 2D line plots in (a)(b)(c) show the performance with changes in sampling variance when channel numbers are 8, 16, 64 from top to bottom. The overall trend of different line plots is similar; however, the inflection points and ranges differ

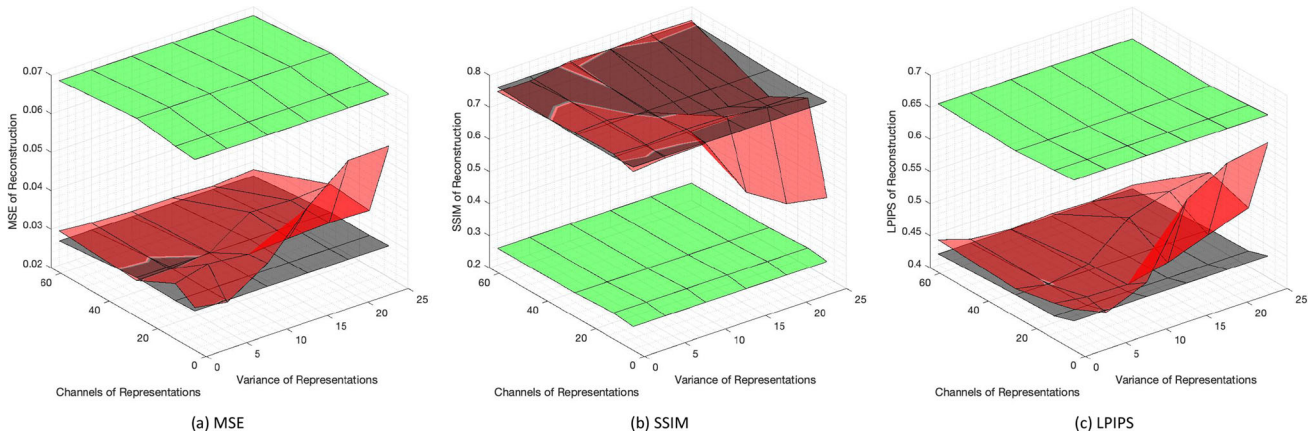

Fig. 11 Effect of frequency sampling distribution on intensity-image reconstruction on Blur-DVS (Jiang et al., 2020). Uniform distribution, "Naive FFT" and Gaussian distribution are highlighted as green, black and red, specifically. The 3D surface plots illustrate the performance, including (a) MSE, (b) SSIM, and (c) LPIPS, with different channel numbers and variances of representations. The "naive FFT" and the Gaussian distribution can obtain comparable reconstruction results, but uniform distribution presents poor reconstruction accuracy

Table 6 Object recognition accuracy compared to existing grid-like representations and state-of-the-art classification algorithms on the N-Cars (Amos et al., 2018) and N-Caltech101 (Orchard et al., 2015)

Algorithm	Classifier	N-Cars	N-Caltech101
HOTS (Lagorce et al., 2016)	SVM	0.624	0.210
HATS (Amos et al., 2018)	SVM	0.902	0.642
HATS (Amos et al., 2018; Gehrig et al., 2019)	CNN	0.909	0.691
H-First (Orchard et al., 2015)	SNN	0.561	0.054
Gabor (Jun Haeng et al., 2016; Neil et al., 2016)	SNN	0.789	0.196
AEGNN (Simon et al., 2022)	GNN	0.945	0.668
Voxel Grid (Rebecq et al., 2019)	CNN	0.933	0.826
EST (Gehrig et al., 2019)	CNN	0.925	0.813
Ours	CNN	0.945	0.847

For a fair comparison, the grid-like representations, including Voxel Grid (Rebecq et al., 2019), EST (Gehrig et al., 2019) and the proposed method, have the same channels $C = 4$ and are fed into the same ResNet-34 (He et al., 2016) network

5.3 Object Recognition

In recent years, event-based classification has gained significant attention for challenging scenarios where conventional cameras may struggle, such as low light conditions and fast object motion. In this section, we use object classification as an example to investigate the performance of the proposed compressed event sensing (CES) volumes on inference tasks.

Following the settings used in Gehrig et al. (2019), we use a pre-trained ResNet-34 (He et al., 2016) as the base network for object prediction. To ensure a fair comparison, events are represented with the same number of channels $C = 4$ using various presentations, including fixed weighted sum-based Voxel Grid (Zhu et al., 2019; Rebecq et al., 2019), learnt weighted sum-based Event Spike Tensor (EST) (Gehrig et al., 2019), and the proposed CES volumes. The networks are implemented using PyTorch (Paszke et al., 2017), and trained using the cross-entropy loss with the ADAM optimizer (Diederi amb Jimmy, 2014) with an initial learning rate of 0.0001, which is halved every 10,000 iterations. The training is conducted for 50 epochs (192,800 and 54,450 iterations for N-Cars and N-Caltech101 datasets, respectively) with a batch size of 4.

Furthermore, we conduct additional comparisons with several state-of-the-art classification algorithms, including two handcrafted representations: HOTS (Lagorce et al., 2016) and HATS (Amos et al., 2018); two baseline implementations of spiking neural networks (SNNs): H-First (Orchard et al., 2015) and Gabor (Jun Haeng et al., 2016; Neil et al., 2016); and one event-based graph neural network (GNN) method: AEGNN (Simon et al., 2022).

Datasets We utilize two public datasets: N-Cars (Amos et al., 2018) and N-Caltech101 (Orchard et al., 2015). The N-Cars dataset is used for the binary classification and contains 12,336 car samples and 11,693 non-cars samples. The event data is recorded by an ATIS event camera (Posch et al., 2010) with a length of 100ms per sample. The N-Caltech101 dataset is an event-based version of the frame-based Caltech101 dataset (Fei-Fei et al., 2006). It mounts an ATIS sensor on a motorized pan-tilt unit, focuses the ATIS on an LCD monitor displaying the original Caltech101 data, and records events as the sensor moves. The N-Caltech101 dataset contains 6,968 samples with 100 object classes plus a background class. We split the training and testing datasets as suggested in Gehrig et al. (2019).

Results and Discussion

Table 6 shows the classification results of our proposed CES volumes compared to other grid-like representations with the same channel number, as well as state-of-the-art methods, on N-Cars and N-Caltech101 datasets.

Compared to Voxel Grid (Rebecq et al., 2019) and EST (Gehrig et al., 2019), our CES volumes outperform them by 1.3%, 2.2% in N-Cars and 2.5%, 4.2% in N-Caltech101. This improvement is attributed to the minimal information loss in our compressed sensing-based approach.

Furthermore, our proposed CES volumes perform favorably against the state-of-the-art classification methods on both datasets. HOTS (Lagorce et al., 2016) and HATS (Amos et al., 2018) which cluster recent events to form a time surface, and thus obscure much temporal information and show less effectiveness for object classification. H-First (Orchard et al., 2015) and Gabor (Jun Haeng et al., 2016; Neil et al., 2016) which feed events into spiking neural networks (SNNs), gain minimal latency but are limited by the learning ability of SNN. The recent event-based GNN method (Simon et al., 2022) preserves the high temporal resolution of events but discards the spatial structure, which limits its effectiveness. In contrast, the proposed CES method can maintain the high temporal information via a compressed sensing scheme and fully exploit the capacity of the convolutional neural network model, resulting in state-of-the-art performance on both N-Cars and N-Caltech101 datasets.

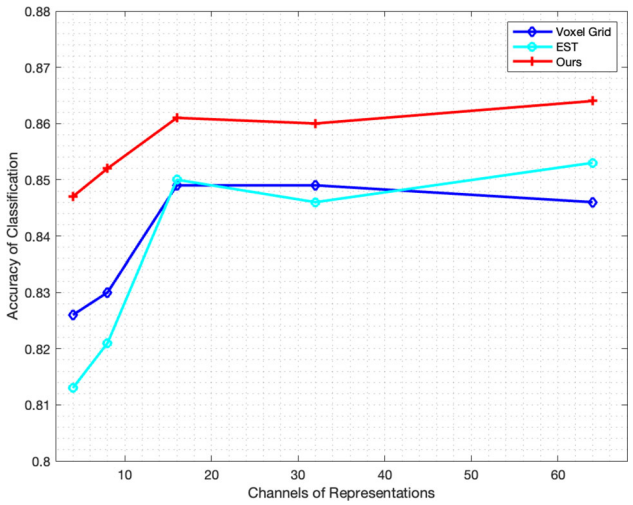

Fig. 12 Effects of channel numbers of representations on object recognition on N-Caltech101 (Orchard et al., 2015). The channel numbers C are set from 4 to 64. The proposed method achieves the highest performance under all numbers of channels and it can perform favorably with few channels, resulting in efficient representation

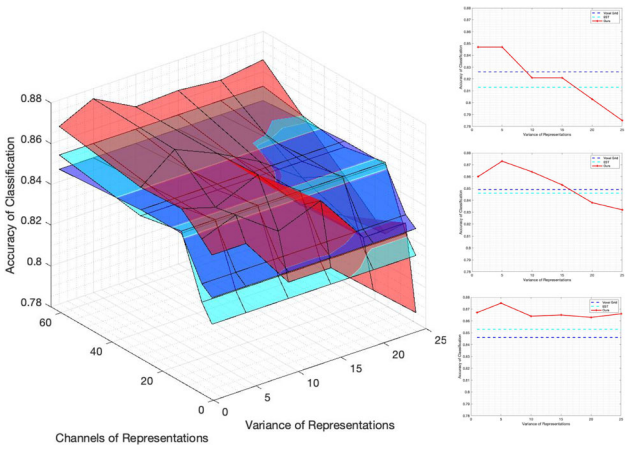

Fig. 13 Comprehensive analysis on the interaction between channel numbers and sampling variances on object recognition on N-Caltech101 (Orchard et al., 2015). Voxel Grid (Rebecq et al., 2019), EST (Gehrig et al., 2019), and the proposed method are highlighted as blue, cyan, and red, specifically. The 3D surface plot illustrate the interaction on accuracy. The channel numbers are set as 4, 8, 16, 32, 64; while frequencies are sampled from a Gaussian distribution with variance σ^2 ranging from 1 to 25. The 2D line plots show the performance with changes in sampling variance when channel numbers are 4, 32, 64 from top to bottom. The overall trend and inflection points of different line plots are similar; however, the ranges differ

Effect of Frequency Number

We further investigate the sensitivity of the frequency numbers on the inference task. We conduct an experiment by setting the number of channels of representations C to different values, specifically $C = 4, 8, 16, 32, 64$, and compare the results with Voxel Grid (Rebecq et al., 2019) and EST (Gehrig et al., 2019) in Fig. 12.

Our method performs favorably across all numbers of channels and the accuracy generally increases with the number of channels. However, we observed that the improvement of accuracy becomes less significant when $C \geq 16$, or equivalently, $M \geq 8$. This suggests that it is not necessary for object recognition tasks to preserve as much accurate temporal information as the event signal reconstruction does. A smaller number of channels are sufficient for achieving good performance, leading to more efficient event data compression. This can be valuable in practical event-driven applications where reducing data size and processing requirements are important considerations.

Effect of Frequency Sampling Range

In our proposed representation, the frequencies for the Fourier transform are tunable. We investigate the effects of the frequency range on the inference task. In this experiment, we set frequency number as 4, 8, 16, 32, 64 and sample frequencies from a Gaussian distribution with variance σ^2 ranging from 1 to 25. The representations are fed into the prediction network and the prediction results are shown in Fig. 13.

Figure 13 illustrates the interaction on object recognition in terms of classification accuracy. We observe that under different channel numbers, the classification accuracy initially improves and then declines with an increase in sampling variance, similar to the results in the intensity-image reconstruction application. However, the inflection points are roughly the same different from the drift appearance in intensity-image reconstruction. One possible reason is that for some low-level computer vision tasks, such as intensity-image reconstruction, dynamic and temporal information play an important role, so that incorporating reasonably high frequency components benefit reconstruction; while for some high-level computer vision tasks, such as object recognition, the structure in spatial domain may be important and high frequency components may disturb and confuse the spatial structure, leading to poor recognition performance.

Effect of Frequency Sampling Distribution

We have discussed the effect of frequency sampling range using Gaussian distribution above, and in this section, we further investigate the impact of different frequency sampling distributions. We conduct an experiment on uniform distribution as well as "naive FFT" which selects the first a few low-order Fourier basis instead of random set of basis.

Figure 14 illustrates the comparison results. As discussed in the effect of frequency sampling range, object recognition is more dependent on lower frequencies and may be disturbed by higher ones. The "naive FFT" method with fixed sampling strategy retains more high frequency information, and thus, exhibits limited improvement with an increase of channel number, and performs less effectively with the large channel number. Uniform distribution cannot ensure the inclusion of

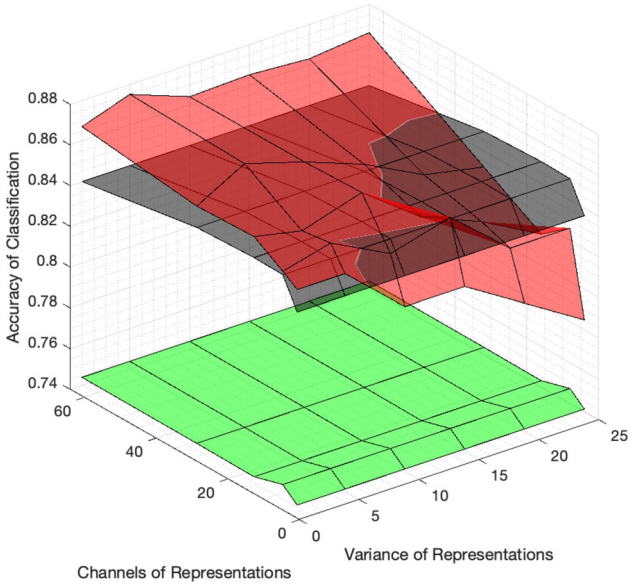

Fig. 14 Effect of Frequency Sampling Distribution on object recognition on N-Caltech101 (Orchard et al., 2015). Uniform distribution, "Naive FFT" and Gaussian distribution are highlighted as green, black and red, specifically. The 3D surface plots illustrate the prediction accuracy with different channel numbers and variances of representations. Gaussian distribution performs favourably with a small sampling variance throughout all channel numbers and gets a stable and large improvement with an increase of channel numbers

enough low-frequency components and thus presents poor classification performance. Instead, Gaussian distribution is more flexible and effective. We find that Gaussian distribution performs favourably with a small sampling variance throughout different channel numbers and get a stable and high accuracy under the large channel numbers. Therefore, we adopt Gaussian distribution throughout the paper.

Overall, determining the optimal sampling distribution is challenging and there is no theoretical justification or guarantees. The optimal frequency sampling may be influenced by different applications as well as the complexity and intrinsic characteristics of event stream.

5.4 Running Time and Latency

One of the key advantages of event cameras is their low latency and high update rate, which makes them suitable for high-speed predictions. To meet the computational demands of event-driven applications, the efficiency of event representations is critical.

We evaluate the computational time of different event representations on the N-Cars testing dataset (Amos et al., 2018) and report the number of processed events per second and the total time used to process a sample of 100ms in Table 7. All representations run on an RTX A5000 GPU. Our proposed CES shares the same computational complexity with other representations. However, due to the efficient implementation using CUDA as stated in Eq. (7), it exhibits more efficiency and is sufficient for most high-speed applications.

5.5 Limitations

"Sim-to-real gap" refers to a degradation in performance when a neural network is trained on simulated data and tested on real data. Currently, there are significant differences between simulated and real events because existing simulators adopt approximate and simple models which cannot comprehensively involves the noise and dynamic effects of event cameras. The frequency properties and distributions of simulated and real events are different. However, our CES operates on frequency domain and encodes more temporal information of the training data. Therefore, in this situation, our advantages may turn into disadvantages.

We conduct an experiment to investigate the influence of "sim-to-real gap" in our proposed CES volumes. Specifically, we focus on the intensity-image reconstruction task. The training dataset is obtained by a simulator ESIM (Rebecq et al., 2018) similar to Rebecq et al. (2019), and the testing dataset is the real Event Camera Dataset (Mueggler et al., 2017).

The results are shown in Table 8. The existing methods, which merge and stack the events into grid-like representations, obscure much temporal information but instead show more robustness to the "sim-to-real gap". Unfortunately, the proposed CES method hinges on frequency domain and encodes more temporal information based on compressed sensing theory. Therefore, the neural networks integrating our CES and trained on simulated data learns the bias towards simulated distribution, resulting in less effectiveness on real data.

It should be emphasized that when the neural networks are trained and tested both on real datasets, our CES takes advantage of its powerful representational capability, resulting in favorable performance against state-of-the-art event representations, as well as high generalization ability across different real datasets. (see Table 5)

6 Conclusion and Future Work

We leverage the sparsity property of events and compressed sensing scheme to show that compressed event sensing (CES) volumes can encode more temporal information of event data, thereby improving event-driven applications. We experimentally show that CES preserves high fidelity and reversibly achieves accurate event reconstruction. We provide theoretical analysis on the expressive power of CES in the deep learning framework. We validate the advantage of CES on a challenging case: dense frame regression test on a

Table 7 Running time for 100ms of event data and number of events processed per second on N-Cars testing dataset (Amos et al., 2018)

Algorithm		Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	Ours
Running time (ms)	↓	1.19	1.53	0.68
Speed (kEv/s)	↑	3415.5	2660.8	5974.7

Table 8 Limitation of CES volumes to the sim-to-real gap. The intensity-image reconstruction algorithms are trained on the simulated event data generated by ESIM (Rebecq et al., 2018) and tested on the real Event Camera Dataset (Mueggler et al., 2017). The existing representations obscure much temporal information and thus are robust to the sim-to-real gap. Unfortunately, the proposed CES volumes, which encode more temporal information, learn the bias towards simulated training distribution, resulting in less effectiveness on real testing dataset

Algorithm		Voxel Grid (Rebecq et al., 2019)	EST (Gehrig et al., 2019)	Ours
Training loss	↓	0.0191	0.0187	**0.0183**
Testing MSE	↓	0.0270	**0.0261**	0.0276
Testing SSIM	↑	**0.5346**	0.5312	0.5267
Testing LPIPS	↓	**0.5302**	0.5463	0.5390

synthetic phantom data. The intensity-image reconstruction and object recognition applications demonstrate that the proposed representation achieves superior performance against the existing representations. Moreover, we thoroughly analyze the effects of Fourier mapping in terms of frequency numbers and frequency selection.

7 Proof of Theorem 1

Theorem 1 *Given a non-zero distinct s-sparse dataset* $\mathfrak{X} = \{\vec{x}_i\}_{i=1}^{I}$, *let* $\mathfrak{H}_a$ *and* $\mathfrak{H}_b$ *be the RKHS associated with the NTK of same-architecture fully-connected network with* $\Psi_a^T\vec{x}$, *and* $\Psi_b^T\vec{x}$ *as input, where* Ψ_a *holds the non-degenerate property while* Ψ_b *does not, the following subset inclusion relation hold:*

$$\mathfrak{H}_b \subsetneq \mathfrak{H}_a. \tag{16}$$

We first introduce two key ingredients of the proof:

Lemma 1 *(Theorem 2.17 in Saitoh et al. (2016)) Let* K_a, K_b $E \times E \to \mathbb{C}$ *be two positive semi-definite kernels. Then the following two statements are equivalent:*

1. *The Hilbert space* $\mathfrak{H}_b$ *is a subset of* $\mathfrak{H}_a$
2. *There exist* $\gamma > 0$, *such that*

$$K_b \preceq \gamma^2 K_a. \tag{17}$$

Lemma 2 *(Proposition 2 in Jacot et al. (2018), Theorem 6 in Luís et al. (2024)) For a fully-connected network adopting a non-polynomial Lipschitz nonlinearity activation function* σ, *for any input dimension* n_0, *the limiting NTK is strictly positive definite if the number of layer* $L \geq 2$.

Proof of Theorem 1 According to Lemma 1, to obtain $\mathfrak{H}_b \subsetneq \mathfrak{H}_a$, we require proof that $\gamma^2 K_a - K_b$ is a positive semidefinite kernel for some $\gamma > 0$, whereas $\gamma^2 K_b - K_a$ is not a positive semidefinite kernel for all $\gamma > 0$.

Consider arbitrary non-empty subset of $\mathfrak{X}$, $\{\vec{x}_i\}_{i=1}^{r} \subset \mathfrak{X}$, for $1 \leq r \leq I$, the NTK matrix $\mathbf{K}$ with size $r \times r$ could be constructed for kernel K_a and K_b, whose entries are

$$\mathbf{K}_a^{i,j} = K_a(\vec{x}_i, \vec{x}_j); \quad \mathbf{K}_b^{i,j} = K_b(\vec{x}_i, \vec{x}_j). \tag{18}$$

As introduced in the proposed NTK model, deep learning methods first represent events using sensing matrix Ψ, and then feed the representation into a neural network g. We refer to the NTK of the network g as $K_g(\cdot,\cdot)$. Therefore, for two different sensing matrices Ψ_a and Ψ_b, the NTK of the whole networks can be represented as

$$\begin{aligned} K_a(\vec{x}_i, \vec{x}_j) &= K_g(\Psi_a^T\vec{x}_i, \Psi_a^T\vec{x}_j) \\ K_b(\vec{x}_i, \vec{x}_j) &= K_g(\Psi_b^T\vec{x}_i, \Psi_b^T\vec{x}_j). \end{aligned} \tag{19}$$

According to Lemma 2, when we adopt the same network settings to Jacot et al. (2018), we obtain that the NTK of the network K_g is a strictly positive definite for distinct network inputs.

Since Ψ_a holds the non-degenerate property as described in Equation (11), the compressed representations $\Psi_a^T\vec{x}_i$ are distinct. Therefore, the NTK matrix $\mathbf{K}_a$ is positive definite with eigenvalues $\lambda_a^i > 0$. Whereas, Ψ_b does not hold this property, i.e., there might exist "degenerate" vector pairs $\vec{x}_i$ and $\vec{x}_j$ such that $\Psi_b^T\vec{x}_i = \Psi_b^T\vec{x}_j$, leading to identical values in i th and j th rows of the NTK matrix $\mathbf{K}_b$. And thus, the eigenvalues $\lambda_b^i \geq 0$.

Therefore, for each non-empty subset $\{\vec{x}_i\}_{i=1}^{r}$ of $\mathfrak{X}$, $\gamma^2\mathbf{K}_a - \mathbf{K}_b$ is positive semidefinite when

$$\gamma = \sqrt{\frac{\max_i \lambda_b^i}{\min_i \lambda_a^i}}. \tag{20}$$

Here, let the γ_{max} be the maximum of γ respective to all the non-empty subsets of $\mathfrak{X}$. Based on the definition of positive semidefinite kernels (see Definition 12.6 in Martin (2019)), $\gamma_{max}^2 K_a - K_b$ is a positive semidefinite kernel on $\mathfrak{X}$. This enables us to apply Lemma 1 to obtain that

$$\mathfrak{H}_b \subseteq \mathfrak{H}_a, \tag{21}$$

Conversely, since the eigenvalues of $\mathbf{K}_b$ contains zero in the degenerate case, for all $\gamma > 0$, $\gamma^2\mathbf{K}_b - \mathbf{K}_a$ is not positive semidefinite. Thus, for all $\gamma > 0$, the kernel function $\gamma^2 K_b - K_a$ is not positive semidefinite. According to Lemma 1, $\mathfrak{H}_a$ is not a subset of $\mathfrak{H}_b$. Combining $\mathfrak{H}_b \subseteq \mathfrak{H}_a$, we can come to $\mathfrak{H}_a \neq \mathfrak{H}_b$, thereby concluding the proof. □

Author Contributions Songnan Lin Conceptualization, Methodology, Software, Writing - original draft preparation; Ye Ma Methodology, Software, Writing - review and editing; Jing Chen Writing - review and editing; Bihan Wen Conceptualization, Writing - review and editing, Supervision;

Funding This work was supported in part by the Ministry of Education, Republic of Singapore, through its Start-Up Grant and Academic Research Fund Tier 1 (RG61/22).

Data Availibility Statement This work does not propose a new dataset. All the datasets we used are publicly available.

Declarations

Conflict of interest The authors have no Conflict of interest to declare that are relevant to the content of this article.

Code availability The code of this work will be released after acceptance.

References

Alonso, I., & Murillo, A. C. (2019). Ev-segnet Semantic segmentation for event-based cameras. In: *CVPRW*. https://doi.org/10.1109/cvprw.2019.00205

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., & Wang, R. (2019). On exact computation with an infinitely wide neural net. *Advances in neural information processing systems,32*.

Bajwa, W. U., Haupt, J., Sayeed, A. M., & Nowak, R. (2010). Compressed channel sensing: A new approach to estimating sparse multipath channels. *Proceedings of the IEEE, 98*(6), 1058–1076. https://doi.org/10.1109/jproc.2010.2042415

Baldwin, R., Liu, R., Almatrafi, M. M., Asari, V. K., & Hirakawa, K. (2022). Time-ordered recent event (tore) volumes for event cameras. *TPAMI*.https://doi.org/10.1109/tpami.2022.3172212

Basarab, A., Liebgott, H., Bernard, O., Friboulet, D., & Kouamé, D. (2013). Medical ultrasound image reconstruction using distributed compressive sampling. In: International symposium on biomedical imaging, pp. 628–631. IEEE. https://doi.org/10.1109/isbi.2013.6556553.

Bi, Y., Chadha, A., Abbas, A., Bourtsoulatze, E., & Andreopoulos, Y. (2019). Graph-based object classification for neuromorphic vision sensing. In: *ICCV*, pp. 491–501. https://doi.org/10.1109/iccv.2019.00058

Bietti, A., & Mairal, J. (2019). On the inductive bias of neural tangent kernels. *Advances in Neural Information Processing Systems*, *32*

Candès, E. J., Romberg, J., & Tao, T. (2006). Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Transactions on Information Theory, 52*(2), 489–509. https://doi.org/10.1109/tit.2005.862083

Candes, E. J., Wakin, M. B., & Boyd, S. P. (2008). Enhancing sparsity by reweighted l 1 minimization. *Journal of Fourier Analysis and Applications, 14*(5), 877–905. https://doi.org/10.1007/s00041-008-9045-x

Carvalho, L., Costa, J. L., Mourão, J., & Oliveira, G. (2024).The positivity of the neural tangent kernel. arXiv preprint arXiv:2404.12928.

Chen, L., & Xu, S. (2020). Deep neural tangent kernel and laplace kernel have the same rkhs. arXiv preprint arXiv:2009.10683.

Chen, Z., Cao, Y., Quanquan, G., & Zhang, T. (2020). A generalized neural tangent kernel analysis for two-layer neural networks. *Advances in Neural Information Processing Systems, 33*, 13363–13373.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory, 52*(4), 1289–1306. https://doi.org/10.1109/TIT.2006.871582

Eldar, Y. C., & Kutyniok, G. (2012). Compressed sensing: theory and applications. *Cambridge University Press*.

Fei-Fei, L., Fergus, R., & Perona, P. (2006). One-shot learning of object categories. *TPAMI, 28*(4), 594–611. https://doi.org/10.1109/tpami.2006.79

Foucart, S., & Rauhut, H. (2013) Restricted isometry property, pp. 133–174. Springer New York, New York, NY.

Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A. J., Jörg Conradt, & Daniilidis, K., et al. (2020). *Event-based vision: A survey. TPAMI, 44*(1), 154–180. https://doi.org/10.1109/TPAMI.2020.3008413

Gehrig, D., Gehrig, M., Hidalgo-Carrió, J., & Scaramuzza, D. (2020). Video to events: Recycling video datasets for event cameras. In: *CVPR,* pp. 3586–3595. https://doi.org/10.1109/cvpr42600.2020.00364

Gehrig, D., Loquercio, A., Derpanis, K. G., & Scaramuzza, D. (2019). End-to-end learning of representations for asynchronous event-based data. In: *ICCV,* pp. 5633–5643. https://doi.org/10.1109/iccv.2019.00573

Gehrig, M., Shrestha, S. B., Mouritzen, D., & Scaramuzza, D. (2020). Event-based angular velocity regression with spiking networks. In: ICRA, pp. 4195–4202. IEEE, https://doi.org/10.1109/icra40945.2020.9197133.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *CVPR, 45*, 770–778. https://doi.org/10.1109/cvpr.2016.90

Hendrycks, D., & Gimpel, K. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, (2016). https://doi.org/10.48550/arXiv.1606.08415.

Huh, D., Sejnowski, T. J. (2018). Gradient descent for spiking neural networks. In: *NeurIPS, 31*. https://doi.org/10.48550/arXiv.1706.04698.

Jacot, A., Gabriel, F., & Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. *NeurIPS, 31*.

Jiang, Z., Zhang, Y., Zou, D., Ren, J., Lv, J., & Liu, Y. (2020) Learning event-based motion deblurring. In: *CVPR*, pp. 3320–3329. https://doi.org/10.1109/cvpr42600.2020.00338.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. ICLR. https://doi.org/10.48550/arXiv.1412.6980.

Lagorce, X., Orchard, G., Galluppi, F., Shi, B. E., & Benosman, R. B. (2016). Hots: a hierarchy of event-based time-surfaces for pattern recognition. *TPAMI, 39*(7), 1346–1359. https://doi.org/10.1109/tpami.2016.2574707

Lee, J. H., Delbruck, T., & Pfeiffer, M. (2016). Training deep spiking neural networks using backpropagation. *Frontiers in Neuroscience, 10*, 508. https://doi.org/10.3389/fnins.2016.00508.

Lin, S., Zhang, J., Pan, J., Jiang, Z., Zou, D., Wang, Y., Chen, J., & Ren, J. (2020). Learning event-driven video deblurring and interpolation. In: ECCV, pp. 695–710. Springer. https://doi.org/10.1007/978-3-030-58598-3_41.

Liu, C., Hui, L. (2023). Relu soothes the ntk condition number and accelerates optimization for wide neural networks. arXiv preprint arXiv:2305.08813.

Maqueda, A. I., Loquercio, A., Gallego, G., García, N., Scaramuzza, D. (2018). Event-based vision meets deep learning on steering prediction for self-driving cars. In: *CVPR*, pp. 5419–5427. https://doi.org/10.1109/cvpr.2018.00568.

Mitrokhin, A., Fermüller, C., Parameshwara, C., & Aloimonos, Y. (2018). Event-based moving object detection and tracking. In: *IROS*, pp. 1–9. IEEE. https://doi.org/10.1109/iros.2018.8593805.

Mitrokhin, A., Ye, C., Fermüller, C., Aloimonos, Y., & Delbruck, T. (2019). Ev-imo: Motion segmentation dataset and learning pipeline for event cameras. In: *IROS*, pp. 6105–6112. IEEE. https://doi.org/10.1109/iros40897.2019.8968520.

Mohtashemi, M., Smith, H., Walburger, D., Sutton, F., & Diggans, J., Sparse sensing dna microarray-based biosensor: Is it feasible? In: *2010 IEEE sensors applications symposium*, pp. 127–130. IEEE (2010). https://doi.org/10.1109/sas.2010.5439412.

Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., & Scaramuzza, D. (2017). 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. https://doi.org/10.1177/0278364917691115

Neil, D., Pfeiffer, M., Liu, S.-C. (2016). Phased lstm: Accelerating recurrent network training for long or event-based sequences. *NeurIPS*, *29*. https://doi.org/10.48550/arXiv.1610.09513.

Nguyen, T. L. N., & Shin, Y. (2013) Deterministic sensing matrices in compressive sensing: A survey. *The Scientific World Journal*, 2013. https://doi.org/10.1155/2013/192795.

Nyquist, H. (1928). Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers, 47*(2), 617–644. https://doi.org/10.1109/5.989875

Orchard, G., Jayawant, A., Cohen, G. K., & Thakor, N. (2015). Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in Neuroscience, 9*, 437. https://doi.org/10.3389/fnins.2015.00437.

Orchard, G., Meyer, C., Etienne-Cummings, R., Posch, C., Thakor, N., & Benosman, R. (2015). Hfirst: A temporal approach to object recognition. *TPAMI, 37*(10), 2028–2040. https://doi.org/10.1109/tpami.2015.2392947

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., & DeVito, Z. , Lin, Z., Desmaison, A., Antiga, L., & Lerer, A. (2017). Automatic differentiation in pytorch.

Posch, C., Matolin, D., & Wohlgenannt, R. (2010). A qvga 143 db dynamic range frame-free pwm image sensor with lossless pixel-level video compression and time-domain cds. *IEEE Journal of Solid-State Circuits, 46*(1), 259–275. https://doi.org/10.1109/jssc.2010.2085952

Rebecq, H., Gehrig, D., Scaramuzza, D. (2018). ESIM: an open event camera simulator. In: *CoRL*.

Rebecq, H., Horstschaefer, T., & Scaramuzza, D. (2017). *Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization.* https://doi.org/10.5244/c.31.16

Rebecq, H., Ranftl, R., Koltun, V., & Scaramuzza, D. (2019). High speed and high dynamic range video with an event camera. *TPAMI, 43*(6), 1964–1980. https://doi.org/10.1109/tpami.2019.2963386

Saitoh, S., Sawano, Y., et al. (2016). *Theory of reproducing kernels and applications*. Springer.

Schaefer, S., Gehrig, D., & Scaramuzza, D. (2022). Aegnn: Asynchronous event-based graph neural networks. In: *CVPR*, pp. 12371–12381. https://doi.org/10.1109/cvpr52688.2022.01205.

Scheerlinck, C., Barnes, N., & Mahony, R. (2018). Continuous-time intensity estimation using event cameras. In: *ACCV*, pp. 308–324. Springer. https://doi.org/10.1007/978-3-030-20873-8_20.

Schölkopf, B., Smola, A. J. (2002). Learning with kernels: Support vector machines, regularization, optimization, and beyond. MIT press.

Seeger, M. (2004). Gaussian processes for machine learning. *International Journal of Neural Systems, 14*(02), 69–106.

Sekikawa, Y., Hara, K., & Saito, H. (2019). Eventnet: Asynchronous recursive event processing. In: *CVPR*, pp. 3887–3896. https://doi.org/10.1109/cvpr.2019.00401.

Sironi, A., Brambilla, M., Bourdis, N., Lagorce, X., & Benosman, R. (2018). Hats: Histograms of averaged time surfaces for robust event-based object classification. In: *CVPR*, pp. 1731–1740. https://doi.org/10.1109/cvpr.2018.00186.

Tancik, M., Srinivasan, P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J., & Ng, R. (2020). Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS, 33*, 7537–7547. https://doi.org/10.1109/mmul.2021.3053698

Wainwright, M. J. (2019). High-dimensional statistics: A non-asymptotic viewpoint, vol. 48. Cambridge university press.

Wang, L., Ho, Y.-S., & Yoon, K.-J. et al. (2019). Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In: *CVPR*, pp. 10081–10090. https://doi.org/10.1109/cvpr.2019.01032.

Wang, Z., Bovik, A. C. Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *TIP, 13*(4), 600–612. https://doi.org/10.1109/tip.2003.819861.

Yang, J., Zhang, Q., Ni, B., Li, L., Liu, J., Zhou, M., & Tian, Q. (2019). Modeling point clouds with self-attention and gumbel subset sampling. In: *CVPR*, pp. 3323–3332. https://doi.org/10.1109/cvpr.2019.00344.

Zhang, H., Chen, X.-H., & Xin-Min, W. (2013). Seismic data reconstruction based on cs and fourier theory. *Applied Geophysics, 10*(2), 170–180. https://doi.org/10.1007/s11770-013-0375-3

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In: *CVPR*, pp. 586–595. https://doi.org/10.1109/cvpr.2018.00068.

Zhang, S., Zhang, Y., Jiang, Z., Zou, D., Ren, J., & Zhou, B. (2020). Learning to see in the dark with events. In: *ECCV*, pp. 666–682. Springer. https://doi.org/10.1007/978-3-030-58523-5_39.

Zhao, B., Ding, R., Chen, S., Linares-Barranco, B., & Tang, H. (2014). Feedforward categorization on aer motion events using cortex-like features in a spiking neural network. *TNNLS, 26*(9), 1963–1978. https://doi.org/10.1109/tnnls.2014.2362542

Zhu, A. Z., & Yuan, L. (2018). Ev-flownet: Self-supervised optical flow estimation for event-based cameras. In: *Robotics: Science and Systems*. https://doi.org/10.15607/rss.2018.xiv.062.

Zhu, A. Z., Yuan, L., Chaney, K., & Daniilidis, K. (2019). Unsupervised event-based learning of optical flow, depth, and egomotion. In: *CVPR*, pp. 989–997. https://doi.org/10.1109/cvpr.2019.00108