DVS-Voltmeter: Stochastic Process-based Event Simulator for Dynamic Vision Sensors

Songnan Lin[1]*, Ye Ma[2]*, Zhenhua Guo[3], and Bihan Wen[1]**

[1] Nanyang Technological University, 50 Nanyang Ave, Singapore
[2] McGill University, Montreal, Canada
[3] Alibaba Group, Hangzhou, China
Code: `https://github.com/Lynn0306/DVS-Voltmeter`

Abstract. Recent advances in deep learning for event-driven applications with dynamic vision sensors (DVS) primarily rely on training over simulated data. However, most simulators ignore various physics-based characteristics of real DVS, such as the fidelity of event timestamps and comprehensive noise effects. We propose an event simulator, dubbed DVS-Voltmeter, to enable high-performance deep networks for DVS applications. DVS-Voltmeter incorporates the fundamental principle of physics - (1) voltage variations in a DVS circuit, (2) randomness caused by photon reception, and (3) noise effects caused by temperature and parasitic photocurrent - into a stochastic process. With the novel insight into the sensor design and physics, DVS-Voltmeter generates more realistic events, given high frame-rate videos. Qualitative and quantitative experiments show that the simulated events resemble real data. The evaluation on two tasks, *i.e.*, semantic segmentation and intensity-image reconstruction, indicates that neural networks trained with DVS-Voltmeter generalize favorably on real events against state-of-the-art simulators.

Keywords: Event Camera; Dataset; Simulation

1 Introduction

Dynamic Vision Sensors (DVS) [17] and related sensors are novel biologically-inspired cameras that mimic human visual perceptual systems. Unlike conventional cameras capturing intensity frames at a fixed rate, DVS respond to brightness changes in the scene asynchronously and independently for every pixel. Once a brightness change exceeds a preset threshold, a DVS triggers an event recording its spatiotemporal coordinate and polarity (sign) of the change. And thus, DVS are endowed with low power consumption, high temporal resolution, and high dynamic range, which attract much attention [9] for challenging scenarios for conventional cameras, such as low latency [24], high-speed motion [19,12],

* Equal contribution

** Corresponding author: bihan.wen@ntu.edu.sg

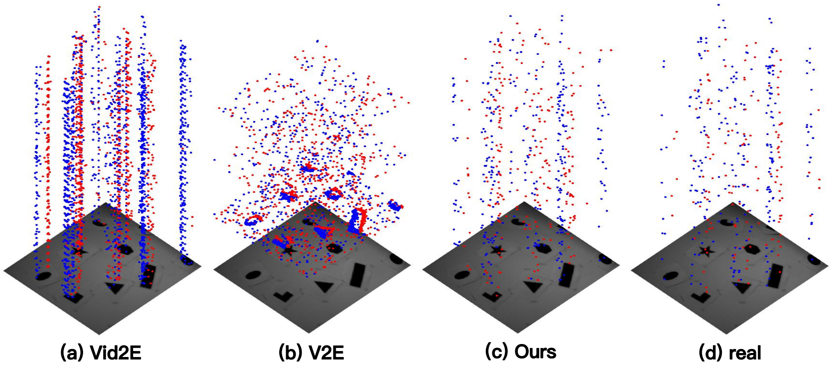


Fig. 1. Visualization of event data and the existing event generation model. (a)-(c) Synthetic events from Vid2E[11], V2E[6], and the proposed DVS-Voltmeter, respectively. (d) Real events. The color pair (red, blue) represents their polarity (1,-1) throughout this paper. Our simulator integrates circuit properties of DVS in a unified stochastic process and can provide more realistic events.

and broad illumination range [27]. Recent works propose to apply deep learning models for event-based vision applications, which have achieved superior results. However, compared to conventional camera images, DVS data are much less accessible and more difficult to obtain. Thus, most of the deep algorithms for event-based applications primarily rely on simulated training data.

DVS simulators utilize the brightness changes calculated from video datasets to simulate event datasets. Existing DVS simulators are like black boxes, modeling the relationship between the brightness changes and the event amount within adjacent frames rather than the attribute changes in the DVS circuit. For example, prior works [20,16] adopt a simple model to determine the event amount by counting the predefined triggering threshold given brightness changes. As they do not consider noise effects, prototyping on simulated data transfers more difficultly to real data. Some attempts have been made to incorporate noises into the simple model. Based on the observation that the triggering threshold is not constant [17], ESIM [23] and Vid2E [11] replace the threshold with Gaussian-distributed one. As shown in Fig. 1 (a), the threshold only varies spatially rather than spatiotemporally, resulting in an apparent artificial pattern with rare noises. Furthermore, V2E [6] incorporates shot noises caused by photon counting into the event model. It randomly adds a certain number of temporal-uniformly distributed noises, which is inconsistent with the distribution of the real ones (see Fig. 1 (b)(d)). Besides, all the algorithms mentioned above adopt linear interpolation to determine the timestamp of events after calculating the event amount between two consecutive frames, resulting in equal-spacing distribution. This simple timestamp sampling strategy inevitably causes overfitting in neural networks and makes them less effective on real data.

In this paper, we provide a new perspective on event simulation from the fundamental voltage properties in the circuit of DVS. Inspired by the conventional-image modeling [7] which approximates the brightness-dependent randomness essentially due to the photon-counting process as Gaussian, we also take the randomness caused by photon reception into consideration and model the voltage signal in DVS as a Brownian motion with drift. Moreover, motivated by [22] which discusses the noise effects of temperature and parasitic photocurrent, we further introduce a Brownian motion term related to temperature and light brightness to simulate noises. Based on the proposed voltage signal model, we develop a practical and efficient event simulator, dubbed DVS-Voltmeter, to generate events from existing videos. We also provide a method to calibrate the model parameters of DAVIS [5,3] which record both DVS events and active pixel sensor (APS) intensity frames. Unlike existing simulators generating events in uniform intervals, the proposed DVS-Voltmeter hinges on the stochastic process, and thus it outputs events at random timestamps. Moreover, as DVS-Voltmeter is based on the circuit principle of DVS, the simulated events resemble real ones (see Fig. 1 (c)(d)) and benefit event-driven applications.

The main contributions of this paper are summarized as follows:

- We offer a novel insight into event modeling based on the fundamental principle of the DVS circuit. Our model utilizes a stochastic process to integrate the circuit properties into a unified representation.
- We propose a practical and efficient event simulator (DVS-Voltmeter) to generate realistic event datasets from existing high frame-rate videos.
- We qualitatively and quantitatively evaluate the proposed simulator and show that our simulated events resemble real ones.
- We validate our simulated events by training neural networks for semantic segmentation and intensity-image reconstruction, which generalize well to real scenes.

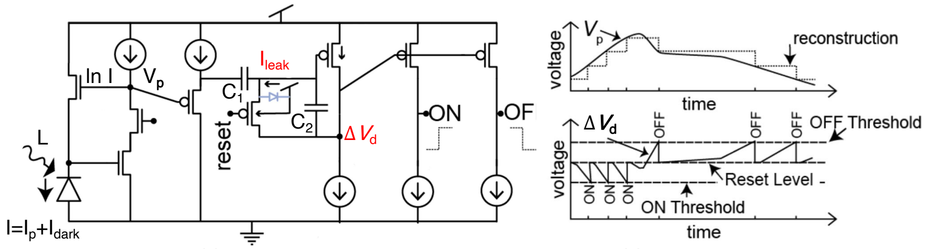
2 DVS Pixel Circuit

Considering that the proposed event simulator hinges on the fundamental voltage properties in the DVS circuit, we revisit the working principle of the DVS circuit to better motivate our event model. As each pixel in DVS independently responds to local brightness changes to generate spike events, we take a pixel circuit of a DVS 128 camera [17] as an example to illustrate the event triggering process.

As shown in Fig. 2 (a), when light signal L hits a pixel on the photoreceptor, it is transduced to a photocurrent with a dark current $I = I_p + I_{dark}$ ($I_p \propto L$) and then logarithmically converted to a voltage V_p. After that, it is amplified to a voltage change $\mathit{\Delta} V_d(t)$ memorized after the last event triggered at the time t_0, which can be ideally formulated as

$$\mathit{\Delta} V_d(t) = -\frac{C_1 \kappa_p U_T}{C_2 \kappa_n}(\ln I(t) - \ln I(t_0)), \tag{1}$$

where C_1, C_2, κ_p, and κ_n are the parameters of the circuit components, and U_T denotes a thermal voltage [17]. Once the DVS detects that $\mathit{\Delta} V_d$ reaches ON



(a) DVS pixel circuit

(b) Operating principle

Fig. 2. Pixel circuit and operation of DVS 128. (a) DVS receives a pixel light signal L and transduces it to a photocurrent I, a voltage V_p, and a voltage change ΔV_d sequentially. (b) Once ΔV_d reaches an ON or OFF threshold, the DVS triggers an ON or OFF event and resets ΔV_d. This figure is adapted from [17,22]. Please see these papers for more details.

threshold $-\Theta_{ON}$ or OFF threshold Θ_{OFF}, it records an ON or OFF event and resets ΔV_d by a pulse as illustrated in Fig. 2 (b), formulated as

$$\begin{cases} \Delta V_d(t) \leq -\Theta_{ON} & \textit{ON events} \\ \Delta V_d(t) \geq \Theta_{OFF} & \textit{OFF events} \\ -\Theta_{ON} < \Delta V_d(t) < \Theta_{OFF} & \textit{no events.} \end{cases} \tag{2}$$

Like conventional cameras, DVS event cameras suffer from complex electromagnetic interference, so it is inappropriate to model the voltage change ΔV_d in the simple way above. As shown in Fig. 2 (a), there is an inevitable junction leakage current named I_{leak}, which affects ΔV_d as

$$\Delta V_d(t) = -\frac{C_1 \kappa_p U_T}{C_2 \kappa_n}(\ln I(t) - \ln I(t_0)) - \int_{t_0}^{t} \frac{1}{C_2} I_{leak}\, du, \tag{3}$$

resulting in a background of activity of ON events.

Prior work [22] validates that the junction leakage current is influenced by the temperature and parasitic photocurrent $I_{leak} = I_{leak\_T} + I_{leak\_pp}$. As for the temperature factor, I_{leak_T} exponentially increases with temperature as

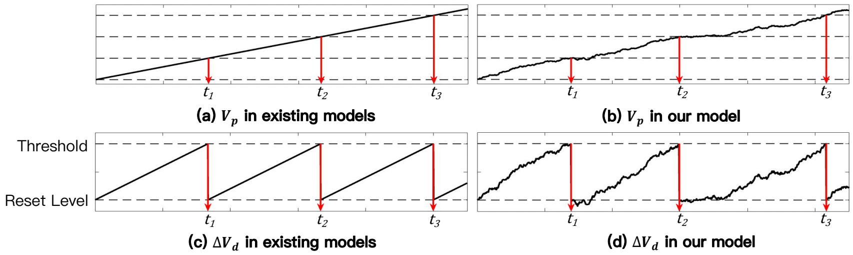
$$I_{leak\_T} \propto \exp(-E_a/k_T T), \tag{4}$$

where E_a is an activation energy, k_T is Boltzmann's constant, and T is the absolute temperature. Meantime, the lighting condition causes parasitic photocurrent like a leakage current

$$I_{leak\_pp} \propto L. \tag{5}$$

3 Stochastic Process-based Event Model

Based on the circuit principle of dynamic vision sensors, we propose to model the time-series voltage change ΔV_d as a stochastic process. We start by constructing a noiseless event model considering the randomness only caused by photon



(a) V_p in existing models

(b) V_p in our model

(c) ΔV_d in existing models

(d) ΔV_d in our model

Fig. 3. Comparison of principles of operation with a same brightness change. (a) Existing models assume that the voltage change $\Delta V_d(t)$ increases linearly, resulting in clean events with equal spaces. (b) The proposed model introduces randomness in timestamps by modeling the process of $\Delta V_d(t)$ as Brownian motion with drift.

reception. And then, we propose a practical model along with the noises caused by the leakage current.

3.1 Noiseless Event Model

We propose to model the voltage change ΔV_d in Eq. (1) considering the randomness only caused by photon reception. During a short period, we assume that local brightness changes linearly with a constant speed k_{dL}. Then, the electrical current ΔI_p transduced from the light signal can be represented by

$$\Delta I_p(t) = k_L k_{dL} \Delta t + N_p(t), \tag{6}$$

where k_L is the transduce rate from light signal to electrical current, $\Delta t = t - t_0$ is a short time after the last event triggered, and $N_p(t)$ is the randomness mainly due to photon reception. The design of $N_p(t)$ is inspired by the model of raw images in conventional cameras [7] that involves a Poissonian component for brightness-dependent randomness essentially due to the photon-counting process. In practice, conventional-image signal processing treats the Poissonian distribution as a special heteroskedastic Gaussian with its variances proportional to brightness. As for the event data, we attempt to model the stochastic process of $N_p(t)$ as a collection of special heteroskedastic Gaussian random variables in the temporal domain. For simplification, we use the Wiener process $W(\cdot)$, also called Brownian motion, to represent the random term by $N_p = \sqrt{m_L \bar{L}} W(\Delta t)$, where m_L is a constant parameter, $\bar{L}$ is the brightness regarded as a constant value within a short time. Thus, the distribution of randomness at every timestamp is Gaussian with its variance proportional to brightness $\bar{L}$.

Then, by combining Eq. (6) and Eq. (1), the voltage change ΔV_d, which determines event triggering, is modeled as a Brownian motion with drift according to the formula[4]:

$$\Delta V_d = -\frac{C_1 \kappa_p U_T}{C_2 \kappa_n} \cdot \frac{1}{I_p(t) + I_{dark}} (k_L k_{dL} \Delta t + \sqrt{m_L \bar{L}} W(\Delta t)). \tag{7}$$

[4] Eq. (7) uses first-order Taylor approximation $\ln I(t) - \ln I(t_0) \approx \frac{1}{I(t)} \Delta I(t)$.

As shown in Fig. 3 (b), our model based on a stochastic process introduces randomness in timestamps and generates events with varying time intervals.

3.2 Event Model with Noises

We further model the voltage change ΔV_d in Eq. (3) considering the noises caused by the leakage current. As discussed above, leakage current is influenced by the temperature and parasitic photocurrent, which affects the voltage ΔV_d and eventually causes noises. According to Eq. (4)(5), we reformulate the leakage current by

$$I_{leak} = m_T \exp(-E_a/k_T) + m_{pp}\bar{L} + N_{leak}, \tag{8}$$

where $m_T > 0$ and $m_{pp} > 0$ are camera-related constants, and N_{leak} is a noise term in the leakage current. Here, we assume N_{leak} is white noise. As the temporal integral of a white noise signal is Brownian Motion $W(\cdot)$ [14], we can rewrite the noise term in Eq. (3) as

$$\Delta V_{d\_leak} = -\frac{1}{C_2}(m_T \exp(-E_a/k_T)\Delta t + m_{pp}\bar{L}\Delta t + \sigma_{leak} W(\Delta t)). \tag{9}$$

Overall, the model for dynamic vision sensors is formulated as

$$\begin{aligned}\Delta V_d =& -\frac{C_1\kappa_p U_T}{C_2\kappa_n} \cdot \frac{1}{I_p + I_{dark}}(k_L k_{dL}\Delta t + \sqrt{m_L \bar{L}} W(\Delta t)) \\ & - \frac{1}{C_2}(m_T \exp(-E_a/k_T)\Delta t + m_{pp}\bar{L}\Delta t + \sigma_{leak} W(\Delta t)) \qquad (10) \\ =& \frac{k_1}{\bar{L}+k_2} k_{dL}\Delta t + \frac{k_3}{\bar{L}+k_2}\sqrt{\bar{L}} W(\Delta t) + k_4\Delta t + k_5\bar{L}\Delta t + k_6 W(\Delta t),\end{aligned}$$

where k_1, k_2, ..., k_6 are calibrated parameters, I_p is replaced by $\bar{L}$ due to their proportional relationship. It can be noticed that the final model can be summarized as a Brownian motion with drift parameter μ and scale parameter σ:

$$\Delta V_d = \mu\Delta t + \sigma W(\Delta t). \tag{11}$$

4 Event Simulation Strategy

As the proposed model considers the event generating process as a Brownian motion with drift, a simple idea to simulate events is to sample from Gaussian distribution at dense timestamps. However, it is computation-intensive, if not impossible, due to the high temporal resolution of event cameras. Therefore, we analyze the property of this stochastic process and propose an efficient event simulator, dubbed DVS-Voltmeter, to decide how and when to trigger events. The overall framework of DVS-Voltmeter is illustrated in Algorithm 1, which alternates between two parts:

- *Polarity Selection:* it determines the polarity of the next triggered event at each pixel based on the hitting probability of the Brownian motion model.
- *Timestamp Sampling:* it samples the timestamp of the next triggered event at each pixel using the first hitting time distribution of Brownian motion.

Algorithm 1 Event Simulation (DVS-Voltmeter).

```
Input: Frames $F_1, F_2, \dots, F_n$ and according timestamps $t_1, t_2, \dots, t_n$
Output: Events generated.
  for each pixel in frames with location $x, y$ do
    Obtain pixel series $P_j = F_j(x, y), j = 1, 2, \dots, n$
    Initialize: $\mathit{\Delta} V_d^{res} \leftarrow 0, t_{now} \leftarrow t_1$
    for $i = 2$ to $n$ do
      $k_{dL} \leftarrow \frac{P_i - P_{i-1}}{t_i - t_{i-1}}, \bar{L} \leftarrow \frac{P_i + P_{i-1}}{2}$
      Compute $\mu$ and $\sigma$ in Eq. (11)
      while $t_{now} < t_i$ do
        Select polarity $p = ON/OFF$ for next event using probability in Eq. (12)
        $\hat{\mathit{\Theta}}_{ON} \leftarrow \mathit{\Theta}_{ON} + \mathit{\Delta} V_d^{res}, \hat{\mathit{\Theta}}_{OFF} \leftarrow \mathit{\Theta}_{OFF} - \mathit{\Delta} V_d^{res}$
        Sample time interval $\tau$ with Eq. (13)(14)
        if $t_{now} + \tau > t_i$ then
          $\mathit{\Delta} V_d^{res} \leftarrow \mathit{\Theta}_p (t_i - t_{now})/\tau$, $t_{now} \leftarrow t_i$
        else
          $\mathit{\Delta} V_d^{res} \leftarrow 0$, $t_{now} \leftarrow t_{now} + \tau$
          Record event with $t_{now}, p, x, y$
        end if
      end while
    end for
  end for
```

Polarity Selection Given a pair of adjacent video frames F_i and F_{i-1} at timestamps t_i and t_{i-1}, respectively, we can obtain an approximated brightness within the capture of two frames by $\bar{L} = (F_i + F_{i-1})/2$ and a brightness changing speed $k_{dL} = \frac{F_i - F_{i-1}}{t_i - t_{i-1}}$. Furthermore, μ and σ in the proposed noise model Eq. (11) can be calculated with a set of well-calibrated parameters of DVS. According to the property of a Brownian motion model, the chance of triggering an ON event next time can be mathematically modeled as the probability of the voltage $\mathit{\Delta} V_d$ hitting $-\mathit{\Theta}_{ON}$ before $\mathit{\Theta}_{OFF}$, formulated as

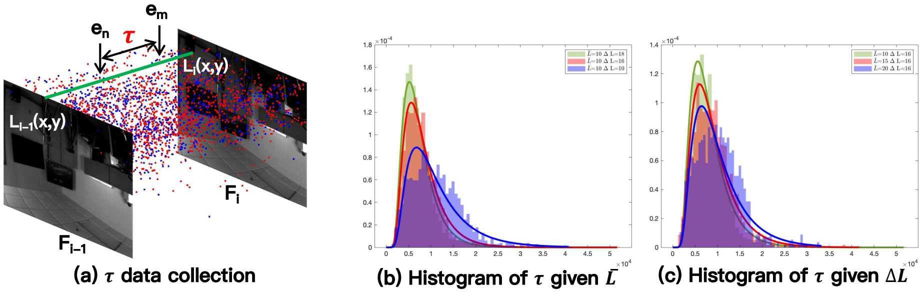
$$P(ON) = \frac{\exp(-2\mu\mathit{\Theta}_{ON}/\sigma^2) - 1}{\exp(-2\mu\mathit{\Theta}_{ON}/\sigma^2) - \exp(2\mu\mathit{\Theta}_{OFF}/\sigma^2)}. \tag{12}$$

And the chance of triggering an OFF event next time is $P(OFF) = 1 - P(ON)$.

Specifically, DVS-Voltmeter performs a uniform sampling within the range [0,1] at each pixel and compares the samples with the corresponding $P(ON)$. As for the pixel where the sample is smaller than $P(ON)$, DVS-Voltmeter triggers an ON event next; otherwise, an OFF one.

Timestamp Sampling After determining the polarity of the next event, we sample its triggering time interval from the distribution of the first hitting time τ of Brownian Motion with Drift. τ follows an inverse Gaussian distribution [8] with non-zero drift parameter μ,

$$\tau_{ON} \sim IG(-\frac{\mathit{\Theta}_{ON}}{\mu}, \frac{\mathit{\Theta}_{ON}^2}{\sigma^2}); \qquad \tau_{OFF} \sim IG(\frac{\mathit{\Theta}_{OFF}}{\mu}, \frac{\mathit{\Theta}_{OFF}^2}{\sigma^2}), \tag{13}$$



(a) τ data collection (b) Histogram of τ given $\bar{L}$ (c) Histogram of τ given ΔL

Fig. 4. Model calibration operation and statistical analysis. (a) Given real captured DAVIS data consisting of frames and events, we collect the time interval τ between two adjacent events at each pixel. τ has an inverse Gaussian distribution / Lévy distribution. (b) The larger the brightness change ΔL, the more compressed the distribution. (c) The larger the average brightness $\bar{L}$, the more spread out the distribution. This statistical result is consistent with the proposed model in Eq. (13)(14).

or a Lévy distribution when $\mu = 0$:

$$\tau_{ON} \sim Levy(\mu, \frac{\Theta_{ON}^2}{\sigma^2}); \qquad \tau_{OFF} \sim Levy(\mu, \frac{\Theta_{OFF}^2}{\sigma^2}). \tag{14}$$

The timestamp sampling uses transformation with multiple roots [18]. For simplification, we set both Θ_{ON} and Θ_{OFF} as 1.

DVS-Voltmeter repeats the polarity selection and timestamp sampling, and updates the timestamp of a new event by $t_{now} = t_{now} + \tau$. The simulator ends until the timestamp of a new event is beyond the frame timestamp t_i. At this moment, we neither record a new event nor update a new timestamp. Instead, we save the residual voltage change ΔV_d^{res} related to the remaining time $t_i - t_{now}$ for follow-up event simulation during the subsequent two adjacent frames.

5 Model Calibration

To train networks generalizing to a specific DVS camera, it is necessary to accurately calibrate k_1, k_2, ..., k_6 in the model in Eq. (10) and generate realistic events for this DVS camera. Ideally, we can look up the camera's specification and conduct a statistical experiment on noise effects to determine the parameters, similar to V2E [6]. However, statistical experiments need complex equipment and, thus, are hard to implement. We provide a calibration method for DAVIS [5,3], which record both events and active pixel sensor (APS) intensity frames.

Specifically, given a sequence of APS frames and corresponding events, for every event recorded between two adjacent frames F_i and F_{i+1}, we can get the brightness conditions when the event occurs, including an approximate brightness $\bar{L} = (F_i + F_{i-1})/2$ and a brightness change $\Delta L = F_i - F_{i-1}$. Furthermore, we collect the time interval τ between this event and the last event triggered at the same pixel (see Fig. 4(a)). Then, given a specific pair of $\bar{L}$ and ΔL, we find the distribution of τ with a form of inverse Gaussian function or Lévy distribution function similar to our assumption in Eq. (13)(14), as shown in Fig. 4(b)(c) and fit it by maximum-likelihood estimation. We further obtain the drift parameter μ and scale parameter σ of the Brownian motion-based event model for each pair of $\bar{L}$ and ΔL. Theoretically, given a set of $\{(\mu_m, \sigma_m, \bar{L}_m, \Delta L_m)\}, m \in \mathbb{N}$, parameters k_1, k_2, ..., k_6 can be calculated by multivariable regression. However, this auto-calibration method is limited by the quality of APS, image quantization, and the assumption of constant brightness changes in our model, so it introduces large errors on σ-related parameters, including k_3 and k_6, in challenging scenes, such as high dynamic range and fast motion. Therefore, we only auto-calibrate μ-related parameters by regression and determine the σ-related parameters manually. Details are provided in our supplementary material.

6 Evaluation

In this section, we provide qualitative and quantitative results on the fidelity of the proposed DVS-Voltmeter and compare it to existing methods [11,6].

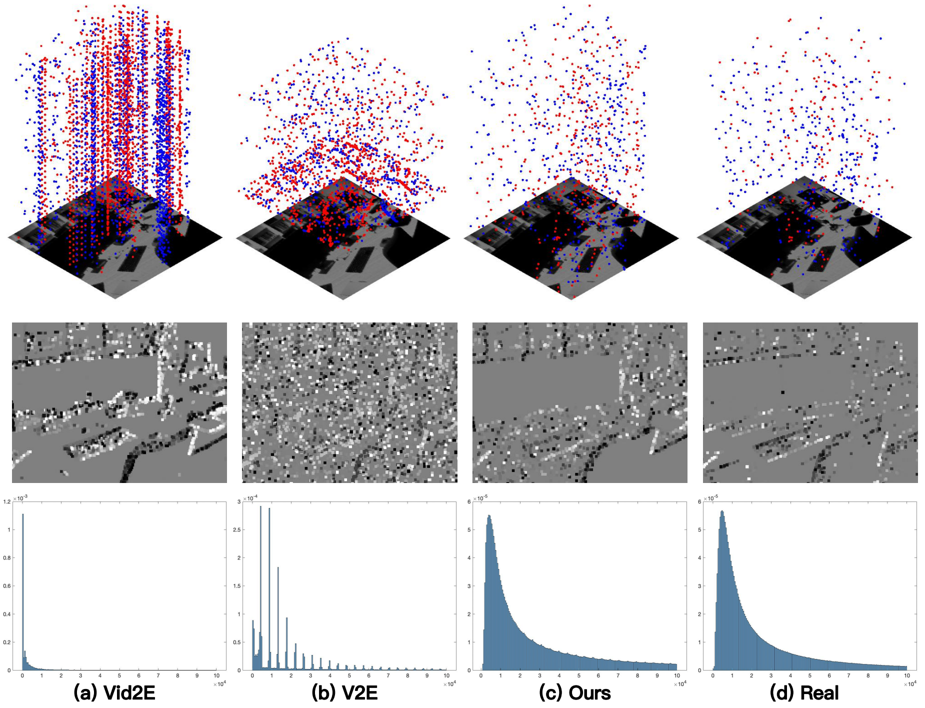
6.1 Qualitative Comparison

We exhibit a side-by-side comparison of a real public DAVIS dataset [20] and its simulated reproductions from Vid2E [11], V2E [6], and our DVS-Voltmeter. For a fair comparison, all simulators firstly interpolate the videos by 10 times to reach a high frame rate using *Super-SloMo* [13], similar to V2E [6]. After that, the simulators generate synthetic events using the interpolated video sequences. Fig. 5 shows the events between two adjacent frames. In addition to illustration in the form of spatiotemporal event clouds, the events are visualized using an exponential time surface [15] with exponential decay of 3.0ms.

As illustrated in Fig. 5, Vid2E [11], which only considers the noises of triggering threshold in DVS, shows an apparent artificial pattern with relatively equal-spacing timestamps. Most events in Vid2E locate around moving edges, resulting in a sharp exponential time surface. Although V2E [6] injects more noises, its strategy of timestamp sampling makes events cluster to limited numbers of time intervals, and most events appear close to the timestamp of frames, leading to unrealistic results. Instead, the proposed DVS-Voltmeter adopts a more flexible timestamp sampling solution based on the stochastic process so that the event clouds spread out and seem realistic. As for the exponential time surface, although there are some differences between the real and simulated events, the result generated from DVS-Voltmeter resembles real data more.

6.2 Effectiveness of Event Model

To quantify the proposed model, we conduct some experiments to compare our simulated events' distribution and noise effects with the statistical results from real DVS. The temperature effect is provided in our supplementary material.

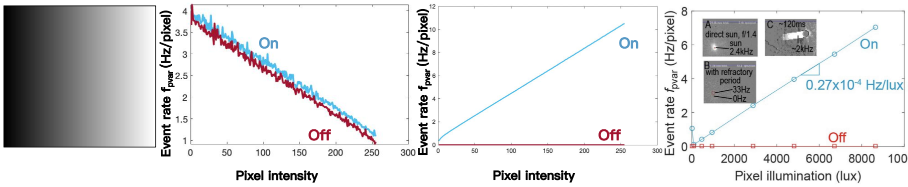


(a) Vid2E (b) V2E (c) Ours (d) Real

Fig. 5. Qualitative and quantitative comparison among Vid2E [11], V2E [6], our DVS-Voltmeter, and real data ‘office_zigzag’ in [20]. We illustrate 3D clouds, 2D time surfaces, and probability density function histograms of event data from top to bottom. Our DVS-Voltmeter gains more randomness, and the generated events resemble real data. More results are provided in our supplementary material.

Event Distribution: To validate the accuracy of event simulators, one might directly compare generated events against the captured ‘ground truth’ events. However, there is no clear metric for the similarity between two event clouds, making the evaluation an ill-posed problem [23]. Therefore, we instead measure the distribution of events. For each event, we calculate the time interval τ after the last event triggered at the same pixel and create a histogram of τ.

Fig. 5 shows the probability density function of the time intervals from the synthetic data and the real one. Vid2E [11] generates events only when encountering brightness changes, and thus, most of the time intervals are within two consecutive interpolated frames (about 4500μs). V2E [6] considers more complex noises but tends to assign events to the timestamps clustered to frames, causing a discrete-like distribution of time intervals. The proposed DVS-Voltmeter is designed with a stochastic process-based model, and thus the time intervals are spread out. Moreover, it hinges on the circuit principle and noise analysis of DVS so that the event distribution of our simulator resembles that of real data more.



(a) Input frame (b) V2E (c) Ours (d) Real

Fig. 6. Comparisons on the effects of brightness-related parasitic photocurrent. We continuously feed a single image (a) into the simulators and measure the noise rates at different intensities. Vid2E [11] does not generate any events. (b)(c)(d) are the results of V2E [6], our DVS-Voltmeter, and real data from DAVIS240C [22]. Our noise rate is similar to the real statistical analysis.

Parasitic Photocurrent: As mentioned in Sec. 2, leak activity caused by parasitic photocurrent increases with light illumination and introduces unintended noises. We measure this noise by continuously inputting a single image with intensity increasing from left to right (see Fig. 6(a)) and counting the noise rate for each intensity.

As there are no brightness changes over the period, previous event simulators [10,20,16,23], including Vid2E [11], do not generate any events. Although V2E [6] considers complex noises, the distribution of generated events (see Fig. 6(b)) is different from the one of real event data provided in [22]. However, our simulator is designed based on the DVS pixel circuit and incorporates a Brownian motion-based noise term to model the noises from parasitic photocurrent, which naturally represents the distribution of real events. As shown in Fig. 6(c), the number of ON events increases with the pixel intensity, and the OFF rate is nearly zero for all points, similar to the real statistical analysis in Fig. 6(d).

7 Example Application

In this section, we validate the proposed simulator on two tasks: semantic segmentation and intensity-image reconstruction. Compared with existing simulators, the deep learning networks trained on our synthetic events perform favorably on real event data.

7.1 Semantic Segmentation

Event-driven semantic segmentation shows the potential for processing challenging scenarios for conventional cameras. In this section, we attempt to train a segmentation network on simulated event datasets and validate its generalization capacity on real data.

Specifically, we reproduce synthetic event datasets from a publicly available DAVIS Driving Dataset (DDD17) [4] captured with a DAVIS346 sensor. As the quality of APS intensity frames limits our model calibration, we utilize the

Table 1. Semantic segmentation performance on the test Ev-Seg data [1] in terms of average accuracy and MIoU (Mean Intersection over Union). The networks are firstly trained on the simulated events and then fine-tuned using 20 real samples.

Training Data	Before Fine-tuning		After Fine-tuning	
	Accuracy	MIoU	Accuracy	MIoU
Vid2E [11]	84.95	46.67	86.41	47.81
V2E [6]	84.11	42.25	84.41	44.32
Ours	**87.88**	**50.60**	**88.51**	**51.20**
Real (20 samples)	67.68	24.39		
Real (All samples)	89.76	54.81		

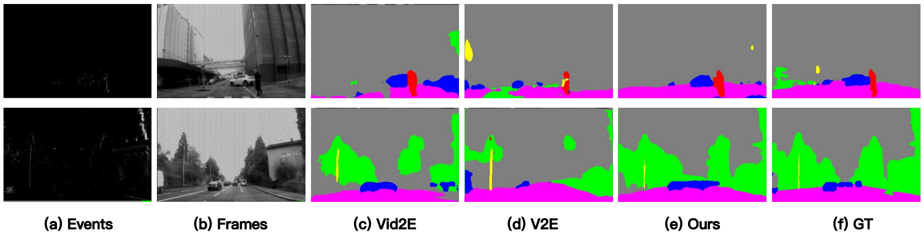


Fig. 7. Visual comparisons on semantic segmentation on Ev-Seg data [1]. The network trained on our simulated events generates more accurate and detailed results. More results are provided in our supplementary material.

'LabSlow' sequence in DVSNOISE20 dataset [2], which is captured with slower and stabler camera movement of a static scene, rather than DDD17 to calibrate DAVIS346 sensors. For a fair comparison, we use the same interpolation strategy on frames in DDD17 by 10 times to generate high frame-rate videos and then generate events by Vid2E [11], V2E [6], and the proposed DVS-Voltmeter. The semantic annotations are provided by [1] for training and testing. The experiment settings, such as event representation, network architecture, and training details, are the same as [1].

We evaluate accuracy and MIoU (Mean Intersection over Union) on semantic segmentation in Table 1. Although the network trained on our simulated data presents slightly lower accuracy than that trained on the whole real event data directly, it performs favorably against state-of-the-art simulators. Fig. 7 provides some examples in the testing set. Our method can give a more accurate and detailed segmentation, which indicates the good resemblance between our events and real ones.

Moreover, we fine-tune the networks given a small-scale real training dataset containing 20 samples for further performance improvement. As shown in Table 1, the network trained only on 20 real samples is overfitted and cannot perform accurately on the testing dataset. However, pre-training on synthetic data and fine-tuning on limited real samples can avoid overfitting and generalize well when testing. Compared to other simulators, our method achieves the highest quantitative results. And it shows a comparable result with the model trained with a large-scale real dataset (All samples). Therefore, using our simulator is

Table 2. Intensity-image reconstruction performance on the Event Camera Dataset [20] in terms of mean squared error (MSE), structural similarity (SSIM) [25], and the calibrated perceptual loss (LPIPS) [26].

	MSE ↓			SSIM ↑			LPIPS ↓		
	Vid2E [11]	V2E [6]	Ours	Vid2E [11]	V2E [6]	Ours	Vid2E [11]	V2E [6]	Ours
dynamic_6dof	0.093	0.177	**0.052**	0.365	0.231	**0.430**	**0.367**	0.437	0.405
boxes_6dof	0.044	0.112	**0.033**	0.509	0.281	**0.521**	0.474	0.590	**0.465**
poster_6dof	0.075	0.158	**0.044**	0.433	0.227	**0.495**	**0.354**	0.511	0.371
shapes_6dof	0.020	0.053	**0.007**	0.707	0.634	**0.790**	0.352	0.375	**0.275**
office_zigzag	0.057	0.125	**0.035**	0.427	0.232	**0.464**	0.507	0.597	**0.483**
slider_depth	0.048	0.108	**0.030**	0.406	0.336	**0.458**	0.523	0.558	**0.501**
calibration	0.051	0.115	**0.036**	0.541	0.393	**0.550**	0.467	0.545	**0.423**
Mean	0.056	0.122	**0.034**	0.505	0.346	**0.550**	0.413	0.501	**0.397**

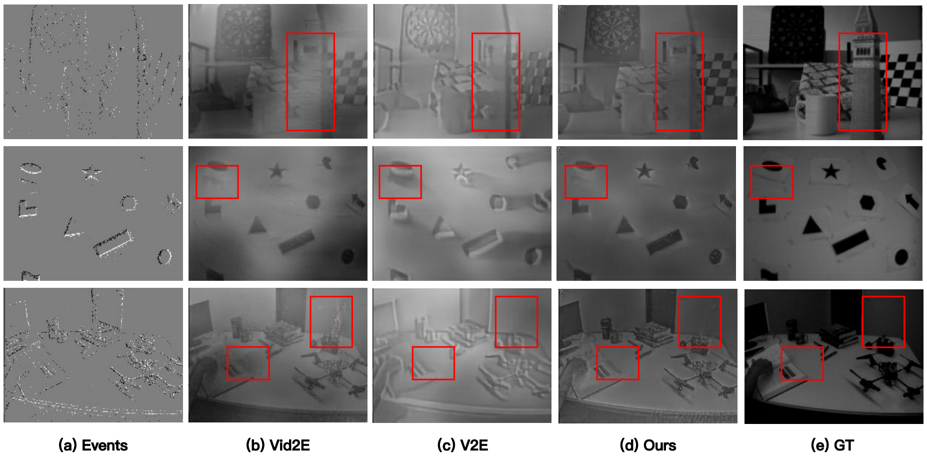


Fig. 8. Visual comparisons on intensity-image reconstruction on Event Camera Dataset [20]. The network trained on our simulated events generates sharper results with fewer artifacts. More results are provided in our supplementary material.

more effective and makes it possible to learn a good segmentation given few or no real training samples.

7.2 Intensity-image Reconstruction

Intensity-image reconstruction aims to generate a high-quality video image from a stream of sparse events, enabling various downstream applications for event-based cameras. Training reconstruction networks requires a large-scale dataset in the form of event streams and the corresponding ground-truth images. However, directly using images captured by DAVIS is inappropriate due to their poor quality, for example, limited dynamic range and blur. Therefore, existing algorithms simulate events from videos to supervise networks.

In this section, we evaluate the effectiveness of our simulator on intensity-image reconstruction by training on synthetic datasets generated from GoPro [21], which provides sharp videos at a frame rate of 240 fps, and testing on

Event Camera Dataset [20] recorded by a DAVIS240C sensor. Specifically, we calibrate the proposed Brownian motion event model for DAVIS240C with the 'office_zigzag' sequence in the testing dataset. Moreover, we use the same frame interpolation strategy to increase the frame rate 10 times and then generate events by Vid2E [11], V2E [6], and the proposed DVS-Voltmeter. Every 1/120 s of events are stacked into a 5-channel spatiotemporal voxel and fed into a recurrent network to reconstruct an image, similar to [24]. The network is trained for 120,000 iterations with a batch size of 2 and a learning rate of 0.0001. Other training details are the same as suggested in [24]. As for testing, because the frame rates of ground truth images in the testing dataset are different among scenes, we generate 4 voxels between two adjacent images and reconstruct 4 × frame-rate videos for quantitative evaluation.

We measure mean squared error (MSE), structural similarity (SSIM) [25], and the calibrated perceptual loss (LPIPS) [26] in Table 2. Our simulator shows better generalization capacity on almost all real datasets with an average 39% decrease in MSE, 9% increase in SSIM, and 4% decrease in LPIPS. Fig. 8 shows some qualitative comparisons side by side. As the proposed simulator is designed based on the statistics and circuit principle of events, it naturally encourages the reconstructed images to have natural image statistics. The results show that the network trained on our simulated events reconstructs more visually pleasing images with finer details and fewer artifacts.

8 Conclusions and Future Work

In this paper, we propose an event model with a novel perspective from the fundamental circuit properties of DVS. The whole model incorporates the voltage variation, the randomness caused by photon reception, and the noises caused by leakage current into a unified stochastic process. Based on the proposed model, we develop a practical and efficient event simulator (DVS-Voltmeter) to generate events from high frame-rate videos. Benefiting from this design, simulated events bear a strong resemblance to real event data. The applications on semantic segmentation and intensity-image reconstruction demonstrate that the proposed simulator achieves superior generalization performance against the existing event simulators.

For future work, one of the main challenges is a more comprehensive characterization of noise effects in DVS, such as temporal noises at low illumination and refractory periods. Besides, a more robust auto-calibration for our model is necessary to mitigate manual calibration.

Acknowledgement

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 (RG137/20).

References

1. Alonso, I., Murillo, A.C.: Ev-segnet: Semantic segmentation for event-based cameras. In: IEEE Computer Vision and Pattern Recognition Workshops (CVPRW). pp. 0–0 (2019)
2. Baldwin, R., Almatrafi, M., Asari, V., Hirakawa, K.: Event probability mask (epm) and event denoising convolutional neural network (edncnn) for neuromorphic cameras. In: IEEE Computer Vision and Pattern Recognition (CVPR). pp. 1701–1710 (2020)
3. Berner, R., Brandli, C., Yang, M., Liu, S.C., Delbruck, T.: A 240× 180 10mw 12us latency sparse-output vision sensor for mobile applications. In: Symposium on VLSI Circuits. pp. C186–C187. IEEE (2013)
4. Binas, J., Neil, D., Liu, S.C., Delbruck, T.: Ddd17: End-to-end davis driving dataset (2017)
5. Brandli, C., Berner, R., Yang, M., Liu, S.C., Delbruck, T.: A 240× 180 130db 3μs latency global shutter spatiotemporal vision sensor. IEEE Journal of Solid-State Circuits **49**(10), 2333–2341 (2014)
6. Delbruck, T., Hu, Y., He, Z.: V2e: From video frames to realistic dvs event camera streams. arXiv preprint arXiv:2006.07722
7. Foi, A., Trimeche, M., Katkovnik, V., Egiazarian, K.: Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. IEEE Transactions on Image Processing (TIP) **17**(10), 1737–1754 (2008)
8. Folks, J.L., Chhikara, R.S.: The inverse gaussian distribution and its statistical application—a review. Journal of the Royal Statistical Society: Series B (Methodological) **40**(3), 263–275 (1978)
9. Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A.J., Conradt, J., Daniilidis, K., et al.: Event-based vision: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **44**(1), 154–180 (2020)
10. Garca, G.P., Camilleri, P., Liu, Q., Furber, S.: pydvs: An extensible, real-time dynamic vision sensor emulator using off-the-shelf hardware. In: IEEE Symposium Series on Computational Intelligence (SSCI). pp. 1–7. IEEE (2016)
11. Gehrig, D., Gehrig, M., Hidalgo-Carrió, J., Scaramuzza, D.: Video to events: Recycling video datasets for event cameras. In: IEEE Computer Vision and Pattern Recognition (CVPR). pp. 3586–3595 (2020)
12. Gehrig, M., Millhäusler, M., Gehrig, D., Scaramuzza, D.: E-raft: Dense optical flow from event cameras. In: International Conference on 3D Vision (3DV). pp. 197–206. IEEE (2021)
13. Jiang, H., Sun, D., Jampani, V., Yang, M.H., Learned-Miller, E., Kautz, J.: Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In: IEEE Computer Vision and Pattern Recognition (CVPR). pp. 9000–9008 (2018)
14. Kuo, H.H.: White noise distribution theory. CRC press (2018)
15. Lagorce, X., Orchard, G., Galluppi, F., Shi, B.E., Benosman, R.B.: Hots: a hierarchy of event-based time-surfaces for pattern recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **39**(7), 1346–1359 (2016)
16. Li, W., Saeedi, S., McCormac, J., Clark, R., Tzoumanikas, D., Ye, Q., Huang, Y., Tang, R., Leutenegger, S.: Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset (2018)

17. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128×128 120db 15μs latency asynchronous temporal contrast vision sensor. IEEE Journal of Solid-State Circuits **43**(2), 566–576 (2008)
18. Michael, J.R., Schucany, W.R., Haas, R.W.: Generating random variates using transformations with multiple roots. The American Statistician **30**(2), 88–90 (1976)
19. Mitrokhin, A., Hua, Z., Fermuller, C., Aloimonos, Y.: Learning visual motion segmentation using event surfaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14414–14423 (2020)
20. Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., Scaramuzza, D.: The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. The International Journal of Robotics Research (IJRR) **36**(2), 142–149 (2017)
21. Nah, S., Hyun Kim, T., Mu Lee, K.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: IEEE Computer Vision and Pattern Recognition (CVPR). pp. 3883–3891 (2017)
22. Nozaki, Y., Delbruck, T.: Temperature and parasitic photocurrent effects in dynamic vision sensors. IEEE Transactions on Electron Devices **64**(8), 3239–3245 (2017)
23. Rebecq, H., Gehrig, D., Scaramuzza, D.: Esim: an open event camera simulator. In: Conference on Robot Learning (CoRL). pp. 969–982. PMLR (2018)
24. Rebecq, H., Ranftl, R., Koltun, V., Scaramuzza, D.: High speed and high dynamic range video with an event camera. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **43**(6), 1964–1980 (2019)
25. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing (TIP) **13**(4), 600–612 (2004)
26. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: IEEE Computer Vision and Pattern Recognition (CVPR). pp. 586–595 (2018)
27. Zhang, S., Zhang, Y., Jiang, Z., Zou, D., Ren, J., Zhou, B.: Learning to see in the dark with events. In: European Conference on Computer Vision (ECCV). pp. 666–682. Springer (2020)