DriveGazen: Event-Based Driving Status Recognition Using Conventional Camera

Xiaoyin Yang, Xin Yang
Dalian University of Technology
tooyoungalex@outlook.com, xinyang@dlut.edu.cn

Abstract

We introduce a wearable driving status recognition device and our open-source dataset, along with a new real-time method robust to changes in lighting conditions for identifying driving status from eye observations of drivers. The core of our method is generating event frames from conventional intensity frames, and the other is a newly designed Attention Driving State Network (ADSN). Compared to event cameras, conventional cameras offer complete information and lower hardware costs, enabling captured frames to encode rich spatial information. However, these textures lack temporal information, posing challenges in effectively identifying driving status. DriveGazen addresses this issue from three perspectives. First, we utilize video frames to generate realistic synthetic dynamic vision sensor (DVS) events.Second, we adopt a spiking neural network to decode pertinent temporal information. Lastly, ADSN extracts crucial spatial cues from corresponding intensity frames and conveys spatial attention to convolutional spiking layers during both training and inference through a novel guide attention module to guide the feature learning and feature enhancement of the event frame. We specifically collected the Driving Status (DriveGaze) dataset to demonstrate the effectiveness of our approach. Additionally, we validate the superiority of the DriveGazen on the Single-eye Event-based Emotion (SEE) dataset. To the best of our knowledge, our method is the first to utilize guide attention spiking neural networks and eye-based event frames generated from conventional cameras for driving status recognition.Please refer to our project page and supplementary materials for more details.

Introduction

Driver state and behavior are crucial to traffic safety. Factors such as the driver's attention level, driving condition, and fatigue directly impact their perception and response to road situations. Developing effective technologies to identify and monitor driver states has become a significant research direction in traffic safety. However, predicting driver states from conventional RGB images is a challenging task; spatial and temporal cues from driving conditions can be adversely affected by head posture and partial occlusion. Existing facial recognition models for classifying driving states in RGB frames are built on complex CNN-based models, such as ResNet 50, Transformer, and Inception-based methods. Different lighting conditions and fast user movements make driver state recognition more complicated, and despite cumbersome large network enhancement modules, driver state recognition from RGB images remains difficult and fragile. We will introduce a novel wearable driver state recognition prototype where users only need to wear a pair of glasses (DG3). Mobile wearable devices can provide stronger feature capture under rapid head movements while offering high resolution for capturing more spatial features and higher temporal resolution for capturing more temporal features. Even though this device provides a stable fixed view of both eyes and traditional camera technology is mature and low-cost, estimating driver states from eye features still faces unique challenges.A key issue is that traditional cameras cannot effectively resolve temporal information under limited lighting conditions. These temporal features are not only crucial for driver state recognition but also important for inferring more informative spatial features. For example, while the eye sockets are major spatial cues, they provide less information for driving state classification. In contrast, subtle movements related to facial units, such as raising the outer eyebrows and squinting, provide stronger cues for eye-based driving state recognition. To address these challenges, we designed the DriveGazen method, which first generates realistic synthetic dynamic visual sensor (DVS) events from video frames and employs the Attention Driving State Network (ADSN) to combine the best features of events and intensity frames, guiding asynchronous event-based driver state recognition with spatial texture cues from the corresponding intensity frames.To train our lightweight eye-based driving state network (ADSN) and stimulate research on event-based eye driver state recognition, we collected a new eye-based event driving state (DriveGaze) dataset. We validated our method on the DriveGaze dataset and demonstrated state-of-the-art driver state recognition capabilities, achieving a significant improvement of 3% in both WAR and UAR compared to the second-best method.

Specifically, our work makes the following five contributions:

- A novel real-time driver state recognition method based on low-cost conventional camera;

- Utilizing video frames to generate realistic synthetic dynamic vision sensor (DVS) events;
- A low-latency spiking neural network with guide attention suited for in-the-wild deployment;
- The first publicly available eye-based event-driven driving state dataset generated from conventional cameras, containing intensity frames and corresponding events, capturing data from different ages, races, genders, etc;
- Validating the superiority of the DriveGazen method on the Single-eye Event-based Emotion (SEE) dataset.

Limitations. DriveGazen partially relies on events generated from intensity frames, which may lead to performance degradation when variations are minimal. While our method effectively handles most scenarios, as confirmed by our experimental results, further improving robustness is an exciting direction for future research in eye-based driving state recognition.

Related Work

We focus on measuring driving status and recognition.Then, we explain the spiking neural network mainly used.

Driving Status Sensing Methods. Researchers utilize physiological measurements such as electroencephalography (EEG), functional near-infrared spectroscopy (fNIRS), electrocardiography (ECG), electrodermal activity (EDA), and respiration (RESP) to identify driver states(Wan et al. 2019). For example, changes in states can lead to variations in facial temperature, hence there are studies employing facial infrared thermal imaging techniques for identification purposes(Zhang, Ihme, and Drewitz 2019). Additionally, some research examines driver states by collecting their hormone levels(Taamneh et al. 2017). Apart from physiological measurements, behavioral measurements are also employed to gauge driver states. Some studies utilize near-infrared (NIR) facial expression recognition methods to identify driver states more accurately(Gao, Yüce, and Thiran 2014). Simultaneously, by analyzing dialogues between drivers and in-vehicle information systems, researchers have found that using voice recognition can identify various states(Jones and Jonsson 2008). Moreover, there are studies that identify driver states through posture movements, validating the feasibility of using radio frequency (RF) technology for state recognition(Raja et al. 2018). In addition to physiological and behavioral measurements, some studies propose collecting information about driver, vehicle status, and changes in the surrounding environment to infer the driver's state(Wan et al. 2019). For instance, based on the pressure characteristics of the throttle and brake pedals, classify the driver's happy and unhappy states(Nor and Wahab 2010). Furthermore, utilizing inertial measurement units (IMUs) to detect driver states is also a common method(Lee et al. 2017). Additionally, some studies use self-report scales to measure driver states, such as Positive and Negative Affect Schedule (PANAS), Self-Assessment Manikin (SAM), and Differential Emotions Scale (DES)(Jeon, Yim, and Walker 2011).

Driving Status Recognition Algorithms. Researchers typically employ supervised machine learning for implementation. Lee et al.(Lee et al. 2017) successfully classified three driving states based on PPG, EMG, and IMU signals using SVM, achieving an accuracy of 99.52%. Ooi et al.(Ooi et al. 2016) and Gao et al.(Gao, Yüce, and Thiran 2014) utilized SVM to classify driver states based on EDA and FEA signals, achieving an accuracy of 85%. Other SVM-based development algorithms are also frequently used for state recognition. For example, Wan et al.(Wan et al. 2019) used Least Squares Support Vector Machine (LS-SVM) to detect states based on multimodal signals. Another commonly used algorithm is k-Nearest Neighbors (kNN); Raja et al. employed this method(Raja et al. 2018) to classify anger and neutral states. Nor and Wahab(Nor and Wahab 2010) used Multi-Layer Perceptron (MLP) to recognize driver states based on velocity and accelerator pedal position. Other traditional machine learning algorithms (such as Bayesian networks) are also used for state recognition(Rebolledo-Mendez et al. 2014). Deep learning algorithms have also been successfully implemented in driver state recognition. Lee et al.(Lee et al. 2018) collected near-infrared and thermal image data of driver's faces and used Convolutional Neural Networks (CNN) to classify driver's anger and neutral states, achieving a recognition accuracy of 99.96%. Although detection accuracy in various studies sometimes reaches 99.96%, most studies are conducted on different datasets. Different recognition tasks, data collection methods, and even different expressions of the same state category can all affect recognition accuracy in driver state detection.

Spiking Neural Network Unlike artificial neural networks (ANN) that are purely digitally coded and whose input and output are numerical values, spiking neural networks (SNN) simulate biological processes, include the concept of time, and only exchange information (pulse), with input and output being pulse sequences. SNN describe the properties of units in the nervous system with varying degrees of detail. SNN simulate three states of biological neurons: resting, depolarized, and hyperpolarized(Ding et al. 2022). When a neuron is in a resting state, its membrane potential remains constant and is usually set to 0. An increase in membrane potential is called depolarization; conversely, a decrease in membrane potential is hyperpolarization. When the membrane potential is above the potential threshold, an action potential, or pulse, is triggered, and a binary-valued pulse signal is used as output to transmit information between neurons. SNN are low-energy biomimetic methods that work in continuous time using discrete signals such as pulses. They can accept the sparsity found in biology and are compatible with high temporal resolution. SNN balance accuracy and computational feasibility. Existing facial driver state recognition methods can usually only identify the peak emotional state or a single driving state in the entire sequence, and are therefore not suitable for applications that require robust estimation of intermediate states. We introduce a lightweight guide attention driving state mothod (DriveGazen) that can effectively recognize various states using SNN. DriveGazen

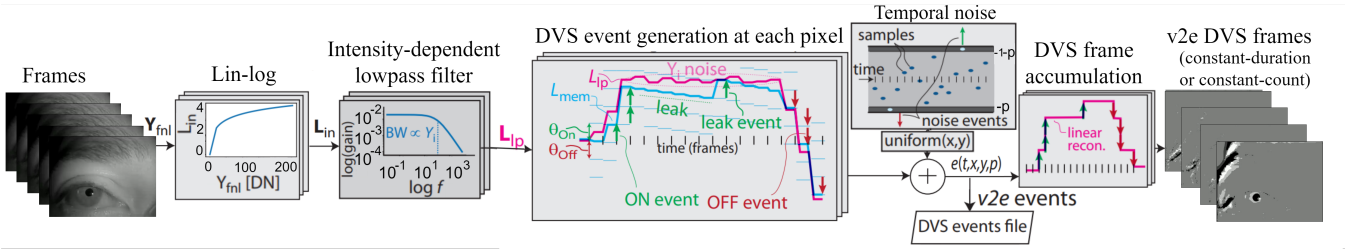

Figure 1: Steps of the v2e DVS event generation (adapted from (Hu, Liu, and Delbruck 2021))

not only remembers the peak phases of individual driving states, but also exploits temporal cues to distinguish different phases, using frames captured by traditional cameras as input and generating event frames. We adopt a hybrid system that utilizes spatial cues and traditional intensity frames to guide temporal feature extraction during training and inference.

DriveGazen

The DriveGazen method first generates realistic synthetic dynamic visual sensor (DVS) events from video frames and utilizes the Attention Driving State Network (ADSN) to combine the best features of events and intensity frames, guiding asynchronous event-based driver state recognition with spatial texture cues from the corresponding intensity frames. Next, I will provide a detailed description of each part of the method.

Video to Event(v2e)

We convert RGB video into grayscale frames, where pixel values are treated as luma intensity values. Figure 1 illustrates the process of synthetic event generation for a single DVS pixel(Hu, Liu, and Delbruck 2021). Y_{fnl} represents the sampled frame. We denote Y as the pixel's luma intensity value in a luma frame Y. Similarly, L represents the pixel's log intensity values in a log intensity frame L.

Standard digital video represents intensity linearly, while DVS pixels detect changes in log intensity. For luma intensity values $Y < 20$ digital numbers (DN), we use a linear mapping from exposure value (intensity) to log intensity to reduce quantization noise in the synthetic DVS output.

Since real DVS pixels have finite analog bandwidth, an optional low-pass filter is used to filter the input L value. The v2e model simulates this effect by making the filter bandwidth (BW) increase monotonically with the intensity value. Although the photoreceptor and source follower form a 2nd-order low-pass filter, one pole usually dominates, so the filter is implemented as an infinite impulse response (IIR) first-order low-pass filter. The nominal cutoff frequency is $f_{3dB}max$ for full white pixels. The filter's bandwidth is proportional to the luma intensity values Y. We denote the filtered L value as L_{lp}. The shape of the filter's transfer function is shown in Figure 1. To avoid nearly zero bandwidth for small DN pixels, an additive constant limits the minimum bandwidth to about 10% of the maximum value.

We assume the pixel has a memorized brightness value L_{mem} in log intensity, and the new low-pass filtered brightness value is L_{lp}. The model then generates a signed integer quantity N_e of positive ON or negative OFF events from the change $\Delta L = L_{lp} - L_{mem}$, where $N_e = \lfloor \Delta L/\theta \rfloor$. If ΔL is a multiple of the ON and OFF thresholds, multiple DVS events are generated. The memorized brightness value is updated by N_e multiples of the threshold.

DVS pixels emit spontaneous ON events called leak events(Nozaki and Delbruck 2017), with a typical rate of approximately 0.1 Hz. These events are caused by junction leakage and parasitic photocurrent in the change detector reset switch(Nozaki and Delbruck 2017). The v2e model adds these leak events by continuously decreasing the memorized brightness value L_{mem}. The leak rate varies according to random fluctuations in the event threshold, decorrelating leak events across different pixels.

The quantal nature of photons leads to shot noise: if, on average, K photons are accumulated in each integration period, then the average variance will also be K. At low light intensities, the effect of shot noise on DVS output events increases significantly, resulting in balanced ON and OFF shot noise events at rates above 1 Hz per pixel. The v2e model simulates temporal noise using a Poisson process. It generates ON and OFF temporal noise events to match a noise event rate R_n (default 1 Hz). To model the increase in temporal noise with reduced intensity, the noise rate R_n is multiplied by a linear function of luma $0 < Y \leq 1$, which reduces noise in brighter areas by a factor $0 < c < 1$ (default $c = 0.25$). This modified rate r is multiplied by the time step Δt to obtain the probability $p = r \times \Delta t \leq 1$, which is applied to the next sample. For each sample, a uniformly distributed number in the range 0-1 is compared against two thresholds $[p, 1 - p]$ to determine if an ON or OFF noise event is generated. These noise events are added to the output and reset the pixels.Please refer to the supplementary materials for more details.

Attention Driving State Network(ADSN)

As illustrated in Figure 2, Specifically, ADSN includes spatial and temporal feature extractors and a guiding attention module. The spatial feature extractor achieves spatial feature extraction by decoupling the sequence length, extracting spatial information only from the first and last frames of the grayscale sequence. ADSN aggregates asynchronous events captured between two grayscale frames into n synchronized event frames. The core of the temporal feature

extractor is the spiking neural layer (SNN), which makes decisions based on membrane potential to remember temporal information from previous event frames. Unlike RNN(Nah, Son, and Lee 2019; Kag and Saligrama 2021), SNN can learn temporal dependencies of arbitrary length without special handling. The guide attention module uses spatial cues from the spatial feature extractor to guide feature learning and enhancement of event frames. It also uses the spiking neural layer to transform spatial features F_s into spikes J_s, which are then fused with the event frames. Next, I will provide a detailed description of each module of the network.

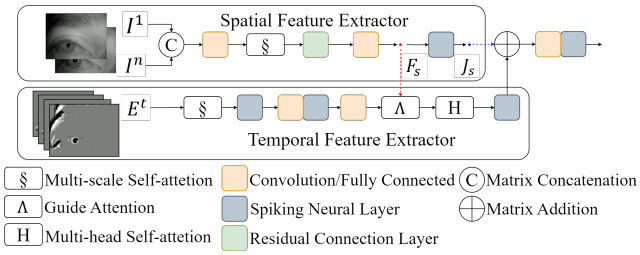

Figure 2: Attention Driving State Network

Spatial Feature Extractor S As illustrated in Figure2, to decouple spatial feature extraction from grayscale sequences in terms of sequence length, the extractor only extracts spatial information from the first frame I_1 and the last frame I_n of the sequence. This not only reduces the reliance on grayscale frames, but also improves recognition performance compared to using all frames, as shown in experiments. First, the two grayscale frames are concatenated in the channel dimension, and then the channel dimension is restored to the original number of channels through a 1×1 convolution kernel, in order to keep consistent with the convolution operation of the temporal feature extractor(Equation 6). Next, multi-scale spatial features are extracted and fused through a multi-scale self-attention module. This allows the network to learn small-scale action unit information and also consider the joint information of larger-scale action units while using residuals to reduce information loss(Equation 2). Then, two 3×3 convolution kernels further extract high-level features in the spatial dimension. In order to better retain the temporal features extracted by the temporal feature extractor, we also use a convolution-spiking neural layer to convert the spatial features F_s into a pulse form J_s(Equation 1), and add it to the temporal features to enhance feature discrimination. Formally, the spatial feature extractor can be defined as:

$$J_s = \Phi^1(F_s) \tag{1}$$
$$F_s = C_3(C_3(\S_{(3,5,7)}(l_s) + l_s)) \tag{2}$$
$$\S(i_1, \ldots, i_n)(\cdot) := C_1\left([\omega_1^s C_{i_1}(\cdot), \ldots, \omega_n^s C_{i_n}(\cdot)]\right) \tag{3}$$
$$\omega_1^s, \ldots, \omega_n^s = \sigma\left([\Upsilon(C_{i_1}(l_s)), \ldots, \Upsilon(C_{i_n}(l_s))]\right) \tag{4}$$
$$\Upsilon(\cdot) := C_1(\varphi(C_1(\mathcal{A}(\cdot)))) \tag{5}$$
$$l_s = C_1([I^1, I^n]) \tag{6}$$

where $(\cdot)$ denotes channel-wise concatenation; C_i and σ denote an $i \times i$ convolutional layer and a softmax function, respectively; $i_1, \ldots, i_n$ denote the value of i in the convolutional layer C_i of the multi-scale self-attention module. According to the equation 2, the values here are 3, 5, 7 respectively. $\mathcal{A}$ denotes the adaptive average pooling layer; φ is a serial operation of a batch normalization operation and a ReLU activation function; Φ^t is a spiking layer that keeps membrane potential from the previous time step, $t-1$. The initial membrane potential, $t = 0$ (see Equation 15).Which only realizes the conversion from floating-point features to 0-1 pulse features without maintaining any temporal information.

Temporal Feature Extractor T The core architecture of the temporal feature extractor is a spiking neural network. Spiking neurons output spike signals based on the accumulation, decay, and reset mechanism of membrane potential to capture the temporal trend in the input sequence. When the membrane potential exceeds a threshold, an action potential (i.e., a spike) is triggered and the membrane potential is reset. The triggering process itself is non-differentiable and cannot be trained by traditional stochastic gradient descent optimization methods. Instead, this paper adopts spatio-temporal backpropagation (STBP) and a convolution-spiking neural layer (Wu et al. 2018) to circumvent this problem. The convolution-spiking neural layer uses a convolution-based layer for signal aggregation and a LIF-based spiking neural layer (Gerstner and Kistler 2002) to manage the potential decay and reset process. This modification makes it possible to learn different accumulation strategies by leveraging convolution-based methods and allows spiking neurons to operate effectively in the temporal domain. The temporal feature extractor receives a total of n event frames as input, denoted as E^1 to E^n, and processes each frame in chronological order. Figure 2 illustrates the architecture of the temporal feature extractor. Formally, after receiving the spatial features F_s and pulse features J_s from the spatial feature extractor, the temporal feature extraction process of E^t can be expressed by equations 7 to 15:

$$O^t = M\left(\tau\left(\tau\left(J_c^t\right)\right)\right) \tag{7}$$
$$J_c^t = \Phi^t(J_e^t) \oplus J_s \tag{8}$$
$$J_e^t = H\left(\Lambda(F_s, F_e^t)\right) \tag{9}$$
$$F_e^t = C_3(\Phi^t(C_3(\Phi^t(\S'_{(3,5,7)}(E^t))))) \tag{10}$$
$$\tau(\cdot) := \Phi^t\left(\Psi(\cdot)\right) \tag{11}$$
$$\S'\left(i_1, \ldots, i_n\right)(\cdot) := C_1\left([\omega_1^s C_{i_1}(\cdot), \ldots, \omega_n^s C_{i_n}(\cdot)]\right) \tag{12}$$

$\S'$ represents the same structure as the multi-scale self-attention module $\S$ in the spatial feature extractor. Ψ is a fully connected layer; M extracts the membrane potential from the spiking neural layer, and Λ is the guide attention module. H is multi-head self-attention. $\Phi^t(\cdot)$ is a spiking neural layer that records the previous spiking state P^{t-1} and accumulated membrane potential V^{t-1}. Upon receiving a new input stimulus X^t, the membrane potential adjusts based on the previous pulse emission and accumulates the new stimulus. The spiking neural layer emits updated pulses P^t and

updates the membrane potential V^t as follows:

$$P^t = h(V^t - \Theta) \tag{13}$$

$$V^t = \alpha V^{t-1}(1 - P^{t-1}) + X^t \tag{14}$$

$$h(x) = \begin{cases} 0 & x < 0 \\ 1 & x >= 0 \end{cases} \tag{15}$$

Θ is the membrane potential threshold, set to 0.3 in this experiment. The parameter α is the attenuation factor for hyperpolarization. The potential V^t is updated such that, for a spike at $t-1$, the membrane potential resets to 0 by scaling $1 - P^{t-1}$, with X^t as the corresponding input. Finally, the driving state is predicted based on the average value of O_t for $t \in [1, n]$, as defined in equation 16:

$$R = \sigma(\frac{1}{n}\sum_{t=1}^{n} O^t) \tag{16}$$

where σ is a Softmax activation function.

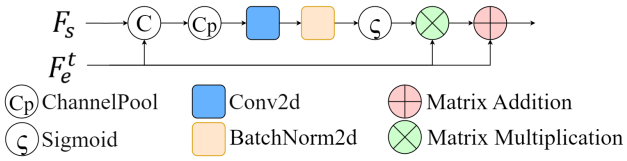

Figure 3: Guide Attention Module

Guide Attention Module Λ Due to the lack of reliable texture information in the event domain, relying solely on event information cannot generate an effective solution. Therefore, we utilize spatial features extracted from grayscale frames to inject rich texture clues into the temporal feature extractor. To guide grayscale frames to guide event frames from the spatial domain to the temporal domain, firstly, the spatial cues F_S learned by the spatial feature extractor are passed through a channel attention module. The learned temporal attention mechanism scores are then allocated to the event frames to enhance the temporal clues. Further spatial attention learning is conducted on the enhanced grayscale frames, and then the spatial attention scores are allocated to the event frames. Finally, the strengthened temporal clues are obtained by adding them to the event frames. The operation process is defined as equations 17 to 20, and the design diagram of the guided attention module is shown in Figure 3.

$$F_{cp} = F_e^t \times F_p + F_e^t \tag{17}$$

$$F_p = \zeta(\varphi(C_1(F_c))) \tag{18}$$

$$F_c = C_P(\Delta) \tag{19}$$

$$\Delta = (F_S) \cdot (F_e^t) \tag{20}$$

Dataset

To the best of our knowledge, there currently does not exist a dataset for driving state recognition based on eye events captured by conventional cameras. To address the lack of training data for event-based driving state recognition, we collected a new event-based driving state dataset (DriveGaze); see Figure 4. DriveGaze consists of driving state eye data from 47 volunteers of different ages, genders, and races, captured using the DG3 eye tracker's conventional camera and converted into event frames; see Figure 1. The DG3 camera is positioned in front of both eyes, with a resolution of 384*288 and a frame rate of 60FPS. Unlike the previously mentioned datasets, our wearable device avoids screen occlusion, resulting in clearer features. Based on conventional cameras, the hardware cost is low, and the application scope is wide.DriveGaze contains original video frames of 7 driving states(see Figure 4). The average length of videos ranges from 30 to 464 frames, with an average of 149.2 frames and a standard deviation of 62.4 frames, reflecting differences in the duration of driving states among different subjects. In total, DriveGaze includes 1645 sequences/245365 frames of original events, with a total duration of 68.1 minutes(Figure 4, divided into 1316 for training and 329 for testing.For more details about the dataset, such as the collection method and environment, category definition, data distribution, etc., please refer to the supplementary materials.

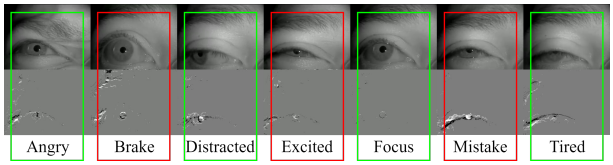

Figure 4: The newly collected Event-based Driving Status (DriveGaze) dataset covers seven classes.

Assessment

Our algorithm is not only to remember the individual's "peak" state, but also to use time clues to distinguish the states of different phases. Therefore, the main goal of this experimental part is to identify any phase of the driving state. When training and testing a state sequence, a uniformly distributed random starting point and the corresponding test time length are selected. The selection of the starting point ensures that the uniformly distributed random starting point keeps the same probability of being selected in any phase of the sequence within the closed interval from the first frame of the sequence to the sequence length minus the test time length. The test time length is defined by the number of event frames used x and the skip time y between two adjacent event frames, denoted as $E_x - S_y$. The skip time defines a window in the time domain where all events are ignored. Without loss of generality, the skip time is expressed as a multiple of $1/60$ s, that is, one frame corresponds to an event frame and a grayscale frame. $E_x - S_y$ means that the test time length is equal to $(x + (x-1) \times y)/60$ seconds. Taking $E_4 - S_3$ as an example, $E_4 - S_3$ means that the number of event frames used in the end is 4; the skip time between adjacent event frames used is $3 \times 1/60$ s, that is, 3 frames are skipped; the test time length is $13/60$ s. Corre-

Methods	2*	Acc. of Driving Status Class (%)							Metrics (%)			
		Ex	Mi	An	Br	Ti	Di	Fo	WAR ↑	UAR ↑	FLOPS (G)	Time (ms)
Resnet18 + LSTM (2016; 1997)	Face	59.1	80.2	58.8	53.9	11.3	81.5	69.1	59.4	60.8	7.9	5.0
Resnet50 + GRU (2016; 2020)	Face	28.5	35.4	45.1	51.6	8.5	70.0	6.5	37.1	36.4	17.3	10.3
3D Resnet18 (2018)	Face	56.0	42.3	61.3	27.6	45.5	42.7	91.2	51.8	52.9	8.3	21.2
R(2+1)D (2018)	Face	64.9	42.4	59.5	32.3	40.8	37.9	93.0	52.4	54.0	42.4	47.3
Former DFER (2021)	Face	83.3	70.1	77.7	68.9	45.6	50.7	90.7	69.4	70.4	8.3	7.7
Eyemotion (2019)	Eye	75.9	79.7	72.0	86.2	84.6	79.0	98.5	83.1	83.3	5.7	17.5
EMO (2020)	Eye	76.7	70.0	63.6	55.8	45.9	54.1	95.6	66.6	66.3	0.3	7.1
SEEN(E4-S3) (2023)	Eye	86.9	83.8	83.6	88.9	88.3	87.6	98.4	88.2	88.2	0.9	7.2
SEEN(E8-S7)	Eye	**93.4**	90.7	82.6	92.2	93.5	87.8	**99.1**	91.3	91.3	0.9	13.4
Ours(E4-S3)	Eye	91.7	83.8	85.7	91.7	92.0	94.5	98.6	91.2	91.2	0.9	7.2
Ours(E8-S7)	Eye	90.5	**91.2**	**90.6**	**93.2**	**94.9**	**94.9**	98.9	**92.4**	**92.4**	0.9	13.4

Table 1: Quantitative comparison retrained and tested on the DriveGaze dataset. The abbreviations are defined as Ex → Excited; Mi → Mistake; An → Angry; Br → Brake; Ti → Tired; Di → Distracted; Fo → Focus. The first and second best results are highlighted in **bold** and underline, respectively.

Methods	2*	Acc. of Emotion Class (%)							Acc. under Light Conditions (%)				Metrics (%)			
		Ha	Sa	An	Di	Su	Fe	Ne	Nor	Over	Low	HDR	WAR ↑	UAR ↑	FLOPS (G)	Time (ms)
Resnet18 + LSTM (2016; 1997)	Face	57.8	86.0	64.9	46.5	9.2	81.6	59.8	57.9	60.4	53.9	52.5	56.3	58.0	7.9	5.0
Resnet50 + GRU (2016; 2020)	Face	27.9	38.0	49.7	44.5	6.9	70.0	5.6	43.0	35.7	28.9	32.8	35.2	34.7	17.3	10.3
3D Resnet18 (2018)	Face	54.8	45.4	67.7	23.8	37.2	42.8	81.6	51.9	51.4	44.8	47.8	49.1	50.5	8.3	21.2
R(2+1)D (2018)	Face	63.6	45.5	65.7	27.8	33.3	37.9	86.6	54.3	50.3	44.4	49.3	49.7	51.5	42.4	47.3
Former DFER (2021)	Face	81.5	75.2	85.8	59.4	39.3	50.8	78.6	70.1	65.4	66.2	61.1	65.8	67.2	8.3	7.7
Former DFER w/o pre-train	Face	44.1	65.2	46.0	66.5	28.0	50.3	36.1	47.0	51.9	45.6	47.2	48.0	48.0	8.3	7.7
Eyemotion (2019)	Eye	74.3	85.5	79.5	74.3	69.1	79.2	94.5	79.0	81.8	**81.5**	72.5	78.8	79.5	5.7	17.5
Eyemotion w/o pre-train	Eye	79.6	85.7	81.2	71.2	54.7	71.6	**96.4**	77.8	75.9	79.8	69.7	75.9	77.2	5.7	17.5
EMO (2020)	Eye	75.0	75.1	70.2	48.1	37.5	54.1	82.8	61.8	62.8	60.1	69.6	63.1	63.3	0.3	7.1
EMO w/o pre-train	Eye	62.0	73.2	60.1	38.7	25.7	48.0	65.3	46.1	60.2	55.5	58.9	53.2	53.3	0.3	7.1
SEEN(E4-S3)(2023)	Eye	**85.0**	89.9	92.2	76.7	72.1	**87.7**	85.2	**83.3**	85.6	80.8	84.8	83.6	84.1	0.9	7.2
SEEN(E7-S1)	Eye	79.0	90.9	91.1	77.2	71.7	85.0	84.4	82.4	86.7	79.8	80.3	82.4	82.7	1.5	10.7
SEEN(E13-S0)	Eye	77.9	88.7	90.2	79.2	69.7	87.6	84.6	81.1	86.5	79.4	81.8	82.3	82.5	2.6	19.0
Ours(E4-S3)	Eye	78.8	**95.0**	**97.1**	**88.3**	**72.1**	75.4	85.0	81.0	**87.6**	80.5	**89.0**	**84.5**	**84.5**	0.9	7.2

Table 2: Quantitative comparison retrained and tested on the SEE dataset. The abbreviations are defined as Ha → Happiness; Sa → Sadness; An → Anger; Di → Disgust; Su → Surprise; Fe → Fear; Ne → Neutrality; Nor → Normal; Over → Overexposure; Low → Low-Light. The first and second best results are highlighted in **bold** and underline, respectively.

spondingly, $E_8 - S_7$ means that the number of event frames used is 8 frames, 7 frames are skipped between frames, and the test time length is $57/60$ s. If the test time length cannot be met due to the short sequence length, the sequence will be read cyclically until it is met. In order to reduce the impact of randomness on the test and ensure the fairness of the comparative experiment, this paper takes randomly selected starting points for all comparative methods for testing, and takes the average of 20 tests after 20 times.We use the same random starting points for single-frame competing methods, where only the random start frame is used.

Metrics

To evaluate the proposed approach and compare it to competing methods, we adopt two widely used metrics: Unweighted Average Recall (UAR) and Weighted Average Recall (WAR) (Schuller et al. 2010). UAR reflects the average accuracy of different driving status classes without considering instances per class, while WAR indicates the accuracy of overall driving status; please refer to the supplementary materials for formal definitions of both metrics.

Training Setup

ADSN is implemented in PyTorch (Paszke et al. 2019). We used Spike-Timing-Dependent Plasticity (STDP) for local weight adjustment and Adam optimizer with the decay rate of the first-order moment estimate set to 0.9, the decay rate of the second-order moment estimate set to 0.999, and the weight decay set to 1×10^{-4} for global optimization of the network. We trained ADSN for 150 epochs using a batch size of 128 on an NVIDIA TITAN V GPU.For the SNN settings, we use a spiking threshold of 0.3 and a decay factor of 0.2 for all SNN neurons.For more details of the experiment please refer to the supplementary materials.

Loss Function

Because driving status recognition is a classification task, we use a regular cross-entropy loss for supervised training of ADSN:

$$\ell = -\frac{1}{7}\sum_{i=1}^{7} y_i \log(\hat{y}_i), \tag{21}$$

where y_i and $\hat{y}_i$ are the predicted i-th probability of driving status and corresponding ground truth probability, respectively.

Evaluation

We compared the effectiveness of DriveGazen with existing recognition methods (including full-face, monocular, and binocular methods) on the collected driving state dataset DriveGaze and the third-party emotion recognition dataset SEE. In order to verify the design ideas of any stage of the recognition state, three eye-based recognition methods Eyemotion(Hickson et al. 2019), EMO(Wu et al. 2020), and SEEN(Zhang et al. 2023) were selected for comparison. In terms of network design, five common face-based temporal information methods were selected to compare with ADSN. They are ResNet18+LSTM(He et al. 2016; Hochreiter and Schmidhuber 1997), Resnet50+GRU(He et al. 2016; Deng, Chen, and Shi 2020), 3D Resnet18(Hara, Kataoka, and Satoh 2018), R(2+1)D(Tran et al. 2018), and Former DFER(Zhao and Liu 2021). For details on training or fine-tuning of each method, please refer to the supplementary materials. Among these previous methods, Eyemotion and EMO are single-frame methods for predicting emotions, while all other methods require full video sequences.We compare DriveGazen with SEEN during different sequence lengths.As shown in Table 1, DriveGazen of E8-S7 provides the best performance of 92.4% and good complexity in driving state recognition.As shown in Table 2, DriveGazen of E4-S3 outperforms the runner-up method SEEN by 1% in WAR and UAR on the emotion recognition dataset SEE. Our method with the same settings also outperforms SEEN by at least 4% in accuracy under overexposure and HDR lighting conditions. Eyemotion performs slightly better than DriveGazen of E4-S3 in low-light conditions. We believe that Eyemotion benefits from being pre-trained on ImageNet(Deng et al. 2009), otherwise Eyemotion's accuracy would be 1% lower than that provided by DriveGaze in the E4-S3 setting. In addition, Eyemotion requires a personalization pre-processing step, which requires subtracting an average neutral image for each person. Personalization significantly improves the accuracy of neutral emotion estimation regardless of whether Eyemotion is pre-trained on ImageNet.

Ablation Study

We perform a series of ablation studies on the DriveGaze dataset that investigate the impacts of input, the influence of each component of ADSN, and the impact of outputs.Table 3 summarizes the experimental results.For more detailed ablation experiments, please refer to the supplementary materials.

Impacts of Input. ADSN leverages the first and last intensity frames. Experiments (A), (B) and (C) gauge the impact of the intensity frames: experiment (A) only uses the first intensity frame, experiment (B) replaces the last intensity frame with the second frame, and experiment (C) uses all the intensity frames corresponding to the included event frames.

		E4-S3		E8-S7	
	Networks	WAR	UAR	WAR	UAR
A	w/o I^n	88.6	88.7	89.7	89.8
B	$I^n \rightarrow I^2$	89.2	89.2	90.3	90.3
C	$[I^1, ..., I^n]$	90.4	90.4	91.5	91.5
D	No Multi-head self-attention	89.5	89.6	90.7	90.9
E	No $F_s \rightarrow J_s$	90.5	90.3	91.6	91.4
F	No Residual	84.0	83.8	85.1	84.9
G	No guide attention	65.4	65.0	66.1	65.8
H	SNN $\rightarrow$ CNN	60.7	60.3	61.4	61.0
I	SNN $\rightarrow$ LSTM	60.8	60.4	61.5	61.1
J	Last potential	88.5	88.6	89.6	89.8
K	Last spike	68.9	68.2	69.8	69.1
M	Ours	91.2	91.2	92.4	92.4

Table 3: Ablation comparisons show that: both the first and last intensity frames are essential for providing discriminative features; all components of ADSN contribute to the overall performance ; and potential averaging is necessary results in a more accurate performance.

Influence of ADSN components. We investigate the effectiveness of the different components that comprise ADSN: 1) the effectiveness of the attention, residual and spatiotemporal features fusion(experiments (D) and (G)) and 2) the benefits of SNNs (experiments (H) to (I)).

Impact of outputs. ADSN estimates status based on the average of n membrane potentials; see Equation 7 and Equation 16. Instead of using the average of n membrane potentials, we define the prediction score based on the potential generated by the last event frame only (experiment (J)); similar to the previous but using output spikes instead of potential (experiment (K)).

Conclusion

We introduce a novel wearable prototype for driver status recognition, which can effectively estimate the driver's status under challenging lighting conditions. We investigated recognition based on input from conventional cameras. Conventional cameras have high resolution, captured frames can robustly encode spatial information. However, parsing temporal cues is challenging. We introduce DriveGazen, a learning-based novel solution for extracting informative temporal cues for status recognition. DriveGazen introduces several novel design components: a method for generating DVS event frames from video frames,a spatial feature extractor based on multi-scale self-attention,a CNN-SNN-based temporal feature extractor and guide attention mechanism. Leveraging spatial awareness and the pulse mechanism of SNN to effectively provide discriminative features for classification. Spatial attention is injected into temporal feature extraction during both training and inference stages. Our extensive experimental results demonstrate that DriveGazen can effectively estimate driver status at any stage. To the best of our knowledge, DriveGazen is the first attempt to utilize conventional camera-generated events and guided attention SNN for driving status recognition tasks.

References

Deng, D.; Chen, Z.; and Shi, B. E. 2020. Multitask emotion recognition with incomplete labels. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, 592–599. IEEE.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Ding, J.; Dong, B.; Heide, F.; Ding, Y.; Zhou, Y.; Yin, B.; and Yang, X. 2022. Biologically inspired dynamic thresholds for spiking neural networks. *Advances in Neural Information Processing Systems*, 35: 6090–6103.

Gao, H.; Yüce, A.; and Thiran, J.-P. 2014. Detecting emotional stress from facial expressions for driving safety. In *2014 IEEE International Conference on Image Processing (ICIP)*, 5961–5965. IEEE.

Gerstner, W.; and Kistler, W. M. 2002. *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press.

Hara, K.; Kataoka, H.; and Satoh, Y. 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 6546–6555.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hickson, S.; Dufour, N.; Sud, A.; Kwatra, V.; and Essa, I. 2019. Eyemotion: Classifying facial expressions in VR using eye-tracking cameras. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1626–1635. IEEE.

Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural computation*, 9(8): 1735–1780.

Hu, Y.; Liu, S.-C.; and Delbruck, T. 2021. v2e: From video frames to realistic DVS events. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1312–1321.

Jeon, M.; Yim, J.-B.; and Walker, B. N. 2011. An angry driver is not the same as a fearful driver: effects of specific negative emotions on risk perception, driving performance, and workload. In *Proceedings of the 3rd international conference on automotive user interfaces and interactive vehicular applications*, 137–142.

Jones, C.; and Jonsson, I.-M. 2008. Using paralinguistic cues in speech to recognise emotions in older car drivers. *Affect and Emotion in Human-Computer Interaction: From Theory to Applications*, 229–240.

Kag, A.; and Saligrama, V. 2021. Time adaptive recurrent neural network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15149–15158.

Lee, B. G.; Chong, T. W.; Lee, B. L.; Park, H. J.; Kim, Y. N.; and Kim, B. 2017. Wearable mobile-based emotional response-monitoring system for drivers. *IEEE Transactions on Human-Machine Systems*, 47(5): 636–649.

Lee, K. W.; Yoon, H. S.; Song, J. M.; and Park, K. R. 2018. Convolutional neural network-based classification of driver's emotion during aggressive and smooth driving using multi-modal camera sensors. *Sensors*, 18(4): 957.

Nah, S.; Son, S.; and Lee, K. M. 2019. Recurrent neural networks with intra-frame iterations for video deblurring. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8102–8111.

Nor, N. M.; and Wahab, A. 2010. Driver identification and driver's emotion verification using KDE and MLP neural networks. In *Proceeding of the 3rd International Conference on Information and Communication Technology for the Moslem World (ICT4M) 2010*, E96–E101. IEEE.

Nozaki, Y.; and Delbruck, T. 2017. Temperature and parasitic photocurrent effects in dynamic vision sensors. *IEEE Transactions on Electron Devices*, 64(8): 3239–3245.

Ooi, J. S. K.; Ahmad, S. A.; Chong, Y. Z.; Ali, S. H. M.; Ai, G.; and Wagatsuma, H. 2016. Driver emotion recognition framework based on electrodermal activity measurements during simulated driving conditions. In *2016 IEEE EMBS conference on biomedical engineering and sciences (IECBES)*, 365–369. IEEE.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, 8024–8035. Curran Associates, Inc.

Raja, M.; Exler, A.; Hemminki, S.; Konomi, S.; Sigg, S.; and Inoue, S. 2018. Towards pervasive geospatial affect perception. *GeoInformatica*, 22: 143–169.

Rebolledo-Mendez, G.; Reyes, A.; Paszkowicz, S.; Domingo, M. C.; and Skrypchuk, L. 2014. Developing a body sensor network to detect emotions during driving. *IEEE transactions on intelligent transportation systems*, 15(4): 1850–1854.

Schuller, B.; Vlasenko, B.; Eyben, F.; Wöllmer, M.; Stuhlsatz, A.; Wendemuth, A.; and Rigoll, G. 2010. Cross-corpus acoustic emotion recognition: Variances and strategies. *IEEE Transactions on Affective Computing*, 1(2): 119–131.

Taamneh, S.; Tsiamyrtzis, P.; Dcosta, M.; Buddharaju, P.; Khatri, A.; Manser, M.; Ferris, T.; Wunderlich, R.; and Pavlidis, I. 2017. A multimodal dataset for various forms of distracted driving. *Scientific data*, 4(1): 1–21.

Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; and Paluri, M. 2018. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 6450–6459.

Wan, P.; Wu, C.; Lin, Y.; Ma, X.; et al. 2019. Driving anger states detection based on incremental association markov blanket and least square support vector machine. *Discrete Dynamics in Nature and Society*, 2019.

Wu, H.; Feng, J.; Tian, X.; Sun, E.; Liu, Y.; Dong, B.; Xu, F.; and Zhong, S. 2020. EMO: Real-time emotion recognition from single-eye images for resource-constrained eyewear devices. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 448–461.

Wu, Y.; Deng, L.; Li, G.; Zhu, J.; and Shi, L. 2018. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12: 331.

Zhang, H.; Zhang, J.; Dong, B.; Peers, P.; Wu, W.; Wei, X.; Heide, F.; and Yang, X. 2023. In the blink of an eye: Event-based emotion recognition. In *ACM SIGGRAPH 2023 Conference Proceedings*, 1–11.

Zhang, M.; Ihme, K.; and Drewitz, U. 2019. Discriminating drivers' emotions through the dimension of power: evidence from facial infrared thermography and peripheral physiological measurements. *Transportation research part F: traffic psychology and behaviour*, 63: 135–143.

Zhao, Z.; and Liu, Q. 2021. Former-DFER: Dynamic Facial Expression Recognition Transformer. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1553–1561.