IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

AMAE: Adaptive Motion-agnostic Encoder for Event-based Object Classification

Yongjian Deng[1] Youfu Li[1] and Hao Chen[2]

***Abstract*— Event cameras, with low power consumption, high temporal resolution, and high dynamic range, have been used increasingly in computer vision. These superior characteristics enable event cameras to perform low-energy and high-response object classification tasks in challenging scenarios. Nevertheless, specific encoding methods for event-based classification are required owing to the unconventional output of event cameras. Existing event-based encoding methods have focused on extracting semantic and motion information in event signals. However, two main problems exist in these methods: (i) the motion information of event signals leads to mispredictions by the classifiers. (ii) effective evaluation methods to validate the motion robustness of event-based classification models have yet to be proposed. In this work, we introduce an adaptive motion-agnostic encoder for event streams to address the first problem. The proposed encoder would allow us to extract indistinguishable semantic information from an object in different motion circumstances. In addition, we propose a novel motion inconsistency evaluation method to assess the motion robustness of the classification models. We apply our method to several benchmark datasets and evaluate it using motion consistency and inconsistency testing methods. Classification performance shows that our proposed encoder outperforms state-of-the-art methods by a large margin.**

I. INTRODUCTION

This study focuses on the object classification problem of event signals captured by neuromorphic event-based cameras. Event-based cameras, such as dynamic vision sensor (DVS)[1] and the ATIS camera[2], work asynchronously on pixel levels rather than trapping in frame-rate limitation. Low power consumption, extremely high temporal resolution (in the order of μs), and high dynamic range(140dB)[3] properties allow event cameras to seek breakthroughs in numerous object classification applications.

Each event from event cameras conveys spatial, temporal locations, and brightness change polarity. These outputs result in extraction difficulties of semantic or motion cues from a single event because each event carries limited information. Therefore, novel encoding methods that convert events into semantic and motion features are urgently needed. Two main encoding method branches can be distinguished based on the different aggregation manners of event messages. The first branch involves encoding methods that can process events asynchronously[4], [5], [6], [7], [8], [9], [10], [11], [12]. These approaches combine newly arrived events and prior knowledge to generate features simultaneously. Thus, these encoders have low latency and are sensitive to changes from each event. In other words, drawbacks exist in expressing robust global semantic information required for classification tasks. The second branch involves encoders that convert a set of events in a predefined interval into feature maps[13], [14], [15], [16]. In contrast to the first branch, this type of design focuses more on extracting semantic and motion information integrally from a packet of events.

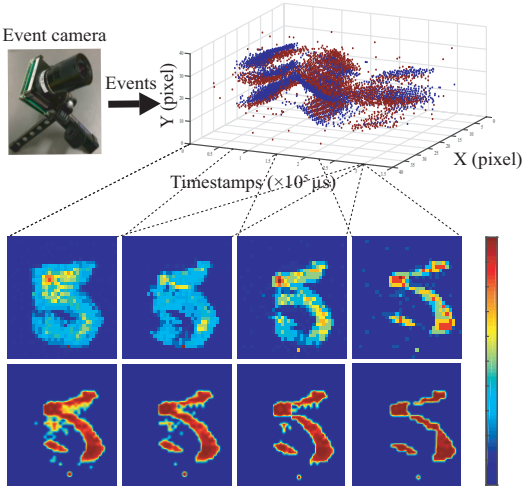


Fig. 1. **Top**: An event-based sensor (left) with its output event signals of number 5 (right), red, and blue dots represents positive and negative events (Fig. 3), respectively. **Middle**: Frames come from integrating the number of positive events with different time intervals. In this integration process, it can be presumed that frames integrated within different time intervals are equivalent to frames generated by various motion trajectories. **Bottom**: Output feature maps from our proposed encoder, AMAE (derived in Sec. III-B), which can extract undistinguished semantic information through the event signals under different conditions.

The encoding methods of both branches have been successfully applied to the task of event-based object classification. However, two main problems in event-based object classification remain understudied. (i) The middle row of Fig 1 shows that event features extracted with different motion trajectories have distinctive semantic features despite being generated from the same object. In other words, the same objects with various motion information will be classified into different classes. (ii) Contrary to complex motion changes in the real world, existing evaluation methods validate event-based classification under the same motion conditions. Thus, the accuracy achieved from these evaluation methods cannot

This work was supported by the Research Grants Council of Hong Kong (Project No. CityU 11203619), and the National Natural Science Foundation of China (Grant No. 61873220). *(Corresponding author: Youfu Li.)*

[1]Yongjian Deng and Youfu Li are with Department of Mechanical Engineering, City University of Hong Kong, Kowloon, Hong Kong {yongjdeng2-c@my., meyfli@}cityu.edu.hk

[2]Hao Chen with the School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore h.chen@ntu.edu.sg

represent models' event-based classification performance under real-world conditions.

In this work, we introduce a novel encoder, namely, the adaptive motion-agnostic encoder (*AMAE*), to address the first problem stated above. This proposed encoder aims to extract undistinguished semantic features of an object under different motion conditions. To achieve this, first, event signals are integrated into frame-based representations, specifically, time-sensitive tensors ($\mathcal{T}_p$), according to their timestamps for different polarities. Next, a differentiable and continuous adaptive motion-agnostic filter (*AMAF*) is proposed. The derived filter based on motion messages conveyed from $\mathcal{T}_p$, emphasizes semantic features and suppresses motion knowledge expression. Adaptive motion-agnostic features ($\mathcal{AMAE}_p$) for object classification are produced (see the bottom row of Fig. 1) after the *AMAF* is applied to $\mathcal{T}_p$. Moreover, we can perform end-to-end event-based object classification owing to the differentiable and continuous properties.

To address the second problem, we propose a novel motion inconsistency evaluation method. This evaluation method is designed based on the fact (stated in Fig 1) that event-based feature maps generated within different time intervals are equivalent to features generated by various motion trajectories. Hence, a model's motion robustness can be evaluated by the classification results of the samples with a random selection of time intervals. Although the existing classification dataset has the limitations of the low diversity of object classes and motion conditions, we believe that the experimental results obtained by this evaluation method are more practical than the traditional evaluation methods.

The main contributions of this paper are summarized as follows:

- Our core contribution is a novel event-based encoder called *AMAE*, which can effectively reduce motion interference in object classification. We can perform end-to-end motion-agnostic event-based object classification by combining this encoder with frame-based deep learning architecture.
- We design a novel motion inconsistency evaluation to assess the motion robustness of event-based classification models.
- Extensive experiments demonstrate the vast superiority of the encoder of the proposed framework over state-of-the-art methods, as well as the advantages of the new evaluation setting.

II. RELATED WORK

Object classification is one of the core problems in computer vision, state-of-the-art performance for this task has improved dramatically in recent years, especially with the help of convolution neural networks' (CNNs) [17], [18]. However, existing methods for traditional images cannot be migrated to an event-based visual system owing to the output form of event cameras. Therefore, novel algorithms for event-based object classification tasks should be designed.

A forthright thought is to build bio-inspired object classification methods to fit the nature of event signals. The most commonly used architecture is derived from SNNs [19], whose structure allows it to process event signals asynchronously. However, SNN-based methods[20], [21], [22], [23] require specialized hardware with durable computational power to complete high-cost back-propagation procedures, which hinders its conversion from theoretical to real-world applications.

Designing a handcraft event signal feature extractor to perform event-based tasks is another option. Motivated by advantages of event cameras, multiple feature extractors have been designed to perform corner detection [24], [25], [26], edge/line extraction [27], [28] and optical flow estimation [4], [29], [30], [31] which are the fundamental tasks of mobile robots. These extractors provide new routes to describe events features, but their performance in complex object classification tasks is not gratified. To accomplish high-level tasks, [11] and [12] proposed event-based representations based on the definition of time surfaces. Specifically, [12] presents an improved version of the method used in [11] by emphasizing the importance of using prior knowledge.The main advantage of these methods is their high sensitivity to motion messages conveyed from each event. However, in addition to this advantage, a common limitation that heavy reliance on the type of motion of the objects in each scene also exists.

To alleviate noise sensitivity and extract global semantic and motion information, recent studies have converted a packet of events into frame-based feature maps and adopted CNNs as their prediction layer [13], [14], [15], [16]. [13] converted events into frames by accumulating different event polarities within a constant temporal window to perform steering-angle prediction. [14] presents four-channel event-based images include polarity and time spikes. [15] introduces the discretized event volume by improving [14] to encode temporal information clearly. Finally, [16] converts event signals into frames by learnable kernels, which reaches state-of-the-art performance in object classification. Frame-based encoding methods are more robust to noise effects and suitable to express global semantic information compared with other approaches. Thus, these types of encoders can substantially improve event-based object classification accuracy. In contrast to previous methods that aim to extract full features from event signals, we propose an encoder that tends to represent the semantic information hidden in events solely and accurately.

Another stream that closely relates to our model is the motion compensation methods [5], [29], [32], [33], [34]. Both our method and motion compensation methods focus on how to decrease the harmful impact of the motion conditions of event signals. The motion compensation methods aim to perform substream tasks with the help of clear gradient images constructed by warping events along the motion trajectories. In contrast, the focus of our method is to improve the classification performance by extracting feature maps with averaged motion information. Our method allows us

to produce feature maps with lower cost compared to the motion compensation methods, which usually contain lots of optimization procedures, although the averaging principle of our approach might result in blurry spatial boundaries, which causes our approach not suitable for pixel-level tasks.

III. METHOD

In this section, we construct the adaptive motion-agnostic encoder for the raw event signals. Our entire event-based classification architecture is demonstrated in Fig. 2.

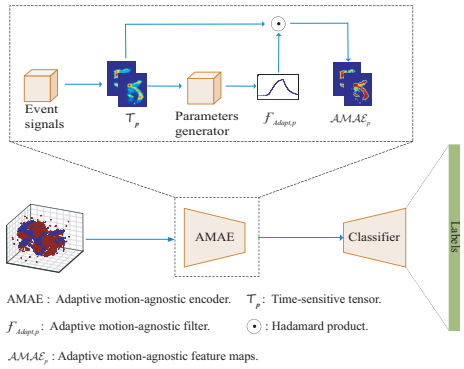


Fig. 2. Our proposed learning architecture for event-based object classification. The parameters generator and Classifier shown in this figure are CNN based learning modules.

A. Event Signals

The pixel of an event camera produce events asynchronously when detecting log brightness $I(x, y, t)$ changes that exceed a nominal threshold C [9]:

$$\Delta \ln I = |\ln I(u, t) - \ln I(u, t - \Delta t)| > C \tag{1}$$

where Δt is the time between the new event and the last event generated at the same location. Specifically, an event $e_i = (x_i, y_i, t_i, p_i)$ will be triggered at pixel (x_i, y_i) at time t_i, where $p \in \{-1, 1\}$ is the sign of the changes of brightness. The working principles of the event camera are shown in Fig. 3. For any given time interval τ, events can be expressed by a sequence:

$$E = \{e_1, \ldots, e_i, e_n\} = \{(x_i, y_i, t_i, p_i)\}_{i=1}^{n} \tag{2}$$

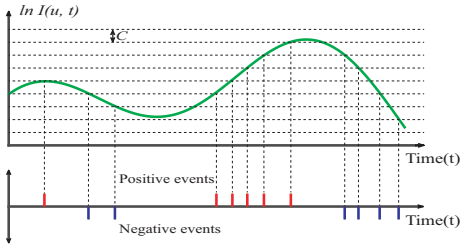


Fig. 3. Interpretation of the working principle of event-based camera. **Top**: Logarithmic changes of the brightness of a pixel. **Bottom**: Events triggered asynchronously corresponding to the nominal threshold C.

Although event signals convey abundant motion and semantic messages with little redundant information, their original form cannot be effectively utilized by artificial neural networks to perform object classification. This is because each event in event signals conveys only a small amount of information, and does not have to be associated with adjacent events due to sensitivity to noise. Therefore, designing an encoder to integrate a set of events to robust and discriminative feature maps is necessary. Specifically, an ideal encoder should be able to mine motion and semantic information in event signals and filter extracted cues to generate representations for classification tasks. For example, feature maps of the same object generated by different motion conditions are similar in an encoder. We refer to this ability as adaptive motion-agnostic, which is one of the most fundamental properties that an event-based classification encoder should have.

B. Adaptive Motion-agnostic Encoder

1) Frame-based representation of event signals: To build an adaptive motion-agnostic encoder, we start by integrating events for each polarity within a given interval to a time-sensitive tensor ($\mathcal{T}_p$) of Eq. (3).

$$\mathcal{T}_{p=\pm 1}(x, y) = \sum_{k}^{N} \delta(x - x_k, y - y_k, p - p_k) t_k^r \tag{3}$$

With the help of *Dirac delta* function δ, in Eq. (3), N events from the time interval τ are accumulated on the two channels frame-based tensor $\mathcal{T}_p$ in terms of the timestamps of event signals. An important observation is that recent events convey more reliable motion information for classification in most cases [29]. For instance, the previous motion messages overlap with the recent motion information, causing the recent semantic information to be blurred, thus cannot be utilized by classifier efficiently. This situation which occurs during fast motion in highly textured scenes, is common. Hence, we stress impacts from recent events by weighting recent events with high values. To this end, we accumulate the timestamps of events as a power function, and define a constant r $(r > 1)$ to measure the importance of recent events.

2) Event-based motion-agnostic filter: First, a motion constant of $\mathcal{T}_p$ has been defined inspired by time images[35], as Eq. (4):

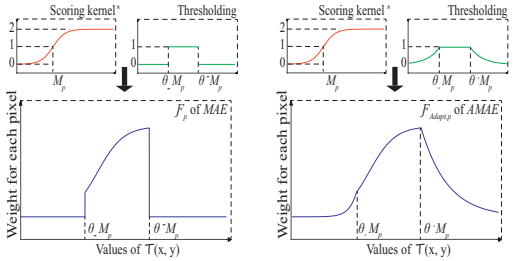
$$\mathcal{M}_p = \frac{\sum_x \sum_y \mathcal{T}_p(x, y)}{\Theta} \tag{4}$$

where Θ denotes the number of non-zero pixels in $\mathcal{T}_p$. In contrast to [35], in which the author used a discretized plane filled with average event timestamps to measure local motion compensation information, our method aims to find a global motion evaluation constant to refine our $\mathcal{T}_p$. Our core idea is to express similar semantic features of objects under different motion conditions. Based on the assumption that the object's motion within a short time interval is constant, the feature maps activated by averaged motion information from various motion trajectories should be similar, although the activated positions in the frame might exist a shift. Therefore, it is helpful to the event-based classification task if we can extract

semantic information with the averaged motion messages of event sequences generated from different motion trajectories. According to the assumption stated above, we generate feature maps with the help of the motion-agnostic filter (*MAF*) and $\mathcal{M}_p$ to weaken the difference in the expression of semantic information among various motion circumstances. We use the symbol $\mathcal{F}_p$ to represent this filter in Eq. (5). This filter ($\mathcal{F}_p \in [0, 2]$ suppresses the interference of inconsistent motion messages in the expression of semantic features by acting as a scoring map to re-weight the $\mathcal{T}_p$.

$$\mathcal{F}_p(x,y) = \begin{cases} 0 & if\ \mathcal{T}_p(x,y) < \theta_- M_p \\ 0 & if\ \mathcal{T}_p(x,y) > \theta^- M_p \\ \tanh\left(\mathcal{T}_p(x,y) - \mathcal{M}_p\right) + 1 & otherwise, \end{cases} \tag{5}$$

Noisy events typically integrated on feature maps with high values, and as we stated in Sec. III-B.1, old events often interfere with the judgment of the classifier. Therefore, we define $\theta_- \in [0, 1)$ and $\theta^- \in [1, +\infty)$ as the lower and upper thresholds in the first two conditions of Eq. (5) to eliminate harmful impacts from noisy and elder events. Notice that θ defined here along with the parameters (λ, β) introduced in Sec. III-B.3 can be different for different polarity frames. For simplicity, the subscripts of these parameters will not include p in this paper. In the last condition of Eq. (5), we have to find a suitable scoring kernel for our filter, which is also the key to our encoding procedure. From Section III-B.1, we generate CNNs friendly event tensors while keeping as many credible messages as possible. Here, according to the assumption stated above in Section III-B.2, we intend to produce objects' feature maps with averaged motion information using those tensors. If we only follow this assumption, then pixels of T_p whose values close to our averaging motion constant M_p should gain higher weight. In this case, $Gaussian\text{-}like(x,\ \mu = M_p)$ functions should be ideal choices to let the representation focus more on averaged motion information. However, based on the characteristic [29] mentioned in Section III-B.1, we would like to let our model give higher weights to the pixels which are incurred by recent events. Under these two constraints, it would be ideal if the kernel allows our frame-based representation to focus on the averaged motion information within the time interval while increasing the impact of recent events. We achieve this by flipping the part which x larger than M_p, along the tangent axis of the peak of the $Gaussian\text{-}like$ function. After this conversion procedure, we obtain a $Tanh\text{-}like$ function, which can give more weights to recent events while retaining the features of the $Gaussian\text{-}like$ function to handle past events. However, the problem is that the representation under this solution might bias too much to the recent motion. We believe that the upper threshold θ^- can also be used to moderate this effect by eliminating the impacts of pixels with overly high values. In this paper, we choose $Tanh$ as our scoring kernel. We apply the *MAF* on the $\mathcal{T}_p$ stated in Eq. (3) to build a complete motion-agnostic encoder (*MAE*) of event signals. The working principle of



(a) (b)

Fig. 4. Comparison of two different filters, (a) $\mathcal{F}_p$ and (b) $\mathcal{F}_{Adapt,p}$. Clear to see the difference between two filters is their threshold kernels. These two different threshold kernels are generated by the first two conditions in (a) and the min function in (b), respectively. Although step-functions used in (a) may perform better as a threshold kernel, the derivative of a step function is 0 means gradient descent will not be able to make progress in updating the weights. In comparison, the threshold kernel in (b) can alleviate this problem.

our proposed *MAE* is summarized in Eq. (6):

$$\mathcal{MAE}_p(x,y) = \mathcal{F}_p(x,y) \times \mathcal{T}_p(x,y) \tag{6}$$

where we define $\mathcal{MAE}_p$ as the motion-agnostic feature maps produced by *MAE*.

3) Adaptive event-based motion-agnostic filter: The *MAE* can be used in different motion environments by manually adjusting its parameters. However, this encoder is not ideal for real-world applications. Complex environmental changes in the real world require encoders to generate targeted feature maps based on obtained input. However, the discontinuous *MAF* module with two wide ranges of zero gradients limits the ability of the *MAE* to perform a back-propagation procedure in learning frameworks. Also, motion biases introduced in the scoring kernel of F_p make the parameter setting more crucial to the classification performance. To alleviate these limitations, we redefine a continuous filter, namely, adaptive motion-agnostic filter (*AMAF*), to allow neural networks to learn to set their parameters better according to input event signals. In the following Eq. (7), We use the symbol $\mathcal{F}_{Adapt,p}$ to represent the *AMAF*.

$$\begin{aligned} &\mathcal{F}_{Adapt,p}(x,y) = \\ &\left(min\left(\exp\left(\lambda(\mathcal{T}_p(x,y) - \theta_- \mathcal{M}_p)\right), 1\right)\right. \\ &+ min(\exp\left(\lambda(-\mathcal{T}_p(x,y) + \theta^- \mathcal{M}_p)\right), 1) - 1) \\ &\left(\tanh(\beta(\mathcal{T}_p(x,y) - \mathcal{M}_p)) + 1\right) \end{aligned} \tag{7}$$

where $\left\{\theta^-, \theta_-\right\}$ have been defined in $\mathcal{T}_p$. In addition, we introduce two new parameters, λ and β, to control the slope of each part of function in $\mathcal{F}_{Adapt,p}$. One thing to note is that two points in $\mathcal{F}_{Adapt}$ are not differentiable, which are $\theta_- \mathcal{M}_p$ and $\theta^- \mathcal{M}_p$. By following the operation introduced in [36], we set the derivative value of these two points as 0. In this way, we achieve the fully differentiable and continuous encoder, the adaptive motion-agnostic encoder (*AMAE*). Differences in the visualization of the $\mathcal{F}_{Adapt,p}$ and $\mathcal{F}_p$ in Fig. 4 are evident. To solve the differences between the step and exponential functions used in filters generation, we

propose λ and β to make $\mathcal{F}_{Adapt,p}$ be more flexible through the change of the slope of the function. We achieve this adaptive procedure with an independent CNNs block, namely parameter generator, which only takes $\mathcal{T}_p$ as input and return parameters $\theta_-, \theta^-, \lambda, \beta$ for different polarity frames. Overall, the procedure to achieve adaptive motion-agnostic feature maps ($\mathcal{AMAE}_p$) of our proposed encoder can be formalized as Eq. (8). This encoder (*AMAE*) can produce feature maps with stressed and undistinguished semantic information from event-based objects under different motion conditions. Fig. 5 shows a comparison of the feature maps generated by our method and other frame-based methods when event signals are generated by different motion conditions.

$$\mathcal{AMAE}_p\left(x,y\right) = \mathcal{F}_{Adapt,p}\left(x,y\right) \times \mathcal{T}_p\left(x,y\right) \tag{8}$$

The corresponding algorithm is given in Algorithm 1.

Algorithm 1: Implementation of adaptive event-based motion-agnostic filter on time-sensitive tensor

Input: $\mathcal{T}_p$
Output: $\mathcal{AMAE}_p$
1. Compute global motion constant using Eq. (4):
   $\mathcal{M}_p \leftarrow computeGlobalMotionConstant(\mathcal{T}_p)$
2. Learn parameters through CNN blocks:
   $\theta_-, \theta^-, \lambda, \beta \leftarrow parametersGenerator(\mathcal{T}_p)$
3. Achieve AMAF through Eq. (7):
   $\mathcal{F}_{Adapt,p} \leftarrow getAMAF(\theta_-, \theta^-, \lambda, \beta, \mathcal{M}_p)$
return $\mathcal{AMAE}_p = \mathcal{F}_{Adapt,p} \bigodot \mathcal{T}_p$

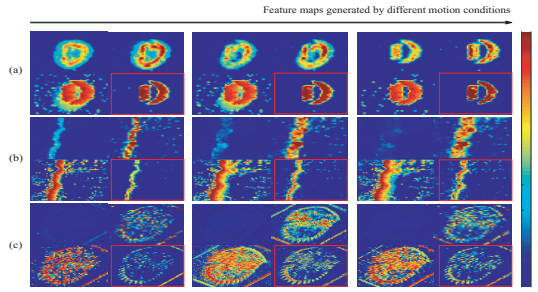


Fig. 5. (a), (b) and (c) show the feature maps generated by the datasets samples under different motion conditions. In this figure, we compare the feature maps obtained by our method with the feature maps obtained by other frame-based methods. For each picture in (a), (b) and (c): top-left: event count frames [13], top-right : the EST method[16], bottom-left: recent timestamps frames[14] , bottom-right: our approach. Compared with other methods, the feature maps generated by our method under different motion conditions retain similar semantic information better, which also reveals that our model is more robust to motion than other methods. As stated in Fig. 1, we only show positive event frames for saving space.

IV. EMPIRICAL EVALUATION

In this section, we perform motion consistency and inconsistency object classification evaluations on the proposed model to validate its classification accuracy and robustness to motion changes. Besides, we analyze the pros and cons of each part of our proposed encoder by conducting ablation studies in different evaluation methods.

A. Datasets

We validate our approach on four different datasets: N-MNIST (N-M)[37], N-CARS (N-C)[12], N-Caltech101 (N-Cal)[37], and CIFAR10-DVS (CIF10). These event-based datasets are generated via two methods. Specifically, N-MNIST, N-Caltech101, and CIFAR10-DVS are obtained by converting conventional images into event signals, and N-CARS is obtained by recording real-world scenes using an ATIS event camera[2]. Due to the different types of recording methods, datasets from converting classic images usually have two limitations. First, this type of recorded method might introduce artificial artifacts to degrade the quality of event streams[37]. Second, conventional images are shot with relatively low dynamic range cameras compared to event cameras; therefore, convert these images to event massages does not fully tap the advantages of event cameras. Although these datasets are born with some defects, they can also reveal which model is more robust to event streams with degraded capture conditions from a side view.

For Datasets that lack an official split. We follow the splitting strategies of previous works for a fair comparison [12], [16] and use 50%, 30%, and 20% of the data as the training, validation, testing set for N-C and using 60%, 20%, and 20% for CIF10. We take the average performance of several random splitting procedures as our final results.

B. Experiment Setup

In this section, we detail how the two evaluation methods are distinguished and analyze the value of our new design evaluation method in practical applications.

1) Motion consistency evaluation: Motion consistency evaluation is a commonly used evaluation method, in which the motion information of the training and the testing sets is the same. In this work, we follow [16] and place the full sequence of each event sample into the encoder at the training and testing stages.

2) Novel motion inconsistency evaluation: The results obtained by the method stated above can provide a standard for evaluating the classification ability of the model. However, this result cannot guarantee the generalization ability of the proposed model because the motion information of events may be substantially different from real-world scenarios. To address this problem, we propose a novel evaluation method to validate the motion robustness of event-based classification models. We keep the same training settings for the motion consistency evaluation but use random duration sequences (approximately $40\% \sim 90\%$ of the entire duration) of each event sample in the datasets for testing. During training, we need to notice that if we augment training set by varying samples' motion conditions, the model's performance will be improved with high probability since this will cause a huge overlap of motion messages between training and testing set. However, the motion inconsistency evaluation is based on that the motion that occurred in the testing set is different from the training set. Therefore, this augmentation fashion will prevent us from measuring models' robustness to unseen motion conditions. Next, we take the averaged

accuracy of several random duration generation procedures as our final result. Given the nature of event cameras, as shown in Fig. 1, and different duration of events produced by different motion trajectories in most cases, this novel sample generation method allows us to simulate the event signals generated in the real world. As for the other interference factors introduced by changing duration, we believe that averaged results can reduce the impact of other factors. Thus, the accuracy results from this evaluation method are credible. Moreover, high accuracy also means higher motion robustness achieved by the testing models.

3) Implementation details: We use ResNet-34 architecture[18] as classifier for each dataset and ResNet18 architecture to learn the *AMAE* parameters. Both architectures are pretrained on traditional RGB images from ImageNet[17]. To adapt the different input and output dimensions between the pretrained and event-based models, we add a channel adaptation layer and an output layer[16] as the first and the last layers of the pretrained model with random weight initialization. We train our networks by optimizing cross-entropy loss with the Adam optimizer[38] for classifiers with learning rate initialized as 10^{-4} (except for the N-MNIST dataset, whose learning rate is set as 10^{-5}) and reducing by a factor of 3.33 every 15 epochs. Moreover, we use Adam optimizer for the *AMAE* with the learning rate initialized as 10^{-5} and reduced by a factor of 5 every 10 epochs. Feature maps from the encoder are resized to 224×224 except N-MNIST (36×36) initially, then fed into the classifier.

C. Event-based Object Classification

In this part, we focus on analyzing the classification results of two evaluation methods between our method and previous works. In our final model, only one parameter r in (3) need to be predefined. Through extensive experimentation, we noticed that the change in r has little effect on accuracy when $r \geq 2$. To prevent the numerical overflow of data types due to excessive values during calculations, we set $r = 2$ for all datasets in this work.

1) Results of motion consistency evaluation: In this section, we firstly compare our results with that of the latest three object classification methods that (i) process events by handcraft feature descriptors, such as HOTS[11] and HATS[12], (ii) perform event-based object classification with SNNs[20], and (iii)classify event-based object through frame-based encoder, such as Voxel Grid[15] and EST[16]. Besides, another method, named motion compensation, to eliminate the impact of motion is compared to our model, such as Lifetime frame[5] and Synchrony frame[29]. In practice, for a fair comparison, we first generate edge frames by motion compensation methods using the same event sequences as our model, and then feed these frames to the resnet34 network to make the classification prediction. The classification results of motion consistency evaluation shown in Table I. From the table, two findings can be inferred as follows. (i) Our approach outperforms state-of-the-art EST[16]. Specifically, our encoder is more suitable for event-based classification than EST because the classification tasks focus more on semantic features and do not require detailed motion information, which contradicts the EST method of extracting as much events' information as possible. (ii) Listed motion compensation methods fail to achieve good results on challenging datasets. Due to their performance on N-C and N-M are comparable with other methods, we think the main reason cause their performance to drop a large margin on N-Cal and CIF10 is their weak robustness to various capture conditions and hyperparameters.

TABLE I
COMPARISON OF CLASSIFICATION ACCURACY ON **MOTION CONSISTENCY** EVALUATION.

Method	N-M	N-Cal	N-C	CIF10
H-First[20]	0.712	0.054	0.561	0.077
Gabor-SNN[12]	0.837	0.196	0.789	0.245
HOTS[11]	0.808	0.21	0.624	0.271
HATS [12]	0.991	0.642	0.902	0.524
HATS + Res34	-	0.691	0.909	-
Lifetime frame[5]	0.899	0.503	0.822	0.197
Synchrony frame[29]	0.984	0.619	0.913	0.285
Voxel Grid[15]	0.987	0.785	0.886	0.731
EST[16]	0.991	0.837	0.925	0.749
AMAE(Ours)	**0.993**	**0.851**	**0.955**	**0.753**

TABLE II
COMPARISON OF CLASSIFICATION ACCURACY ON **MOTION INCONSISTENCY** EVALUATION.

Method	N-M	N-Cal	N-C	CIF10
Lifetime frame[5]	0.801	0.430	0.800	0.163
Synchrony frame[29]	0.927	0.548	0.901	0.237
Voxel Grid[15]	0.871	0.707	0.853	0.653
EST[16]	0.897	0.764	0.908	0.693
AMAE(Ours)	**0.984**	**0.828**	**0.942**	**0.733**

2) Results of motion inconsistency evaluation: Subsequently, we next evaluate the motion robustness of our proposed method using the novel evaluation method. Codes for HATS, Gabor-SNN, and HOTS, have yet to be released publicly. Thus, we compare our method with other frame-based and motion compensation methods in terms of classification accuracy for a fair comparison, as shown in Table II. From the table, we can find that our model and motion compensation methods, while reducing some accuracy, remain at the same level as the motion consistency evaluation. Conversely, the classification performance of Voxel Grid and EST drops by a large margin owing to different motion information interference with the judgment of their classifiers. This comparison reveals that our proposed method is robust for classifying event-based objects generated by different motion conditions, and it also proves the effectiveness of our evaluation method in measuring motion robustness.

D. Ablation Study

In this section, we conduct an ablation study on each dataset to explore the functionality of each component in our proposed encoder. We evaluate our encoder in three forms: $(i)$the encoder that uses only $\mathcal{T}_p$ as the classifier input. (ii) the *MAE* which combines $\mathcal{T}_p$ and $\mathcal{F}_p$. and (iii) the *AMAE* that combine $\mathcal{T}_p$ and $\mathcal{F}_{Adapt,p}$. The training and testing procedures are the same across different encoding types. We set parameters θ^-, and θ_- as 5 and 0.2 for all datasets in the *MAE* and use the encoder that only uses $\mathcal{T}_p$ as our baseline.

1) Ablation study on motion consistency evaluation: Table III shows two phenomena. First, the *MAE* and *AMAE* outperforms $\mathcal{T}_p$, which reveals the effectiveness of our proposed filters. Second, the *MAE* achieves better performance than the *AMAE* in the motion consistency evaluation at most cases. This result may be because the step function used in *MAE* is more adequate than the exponential function adopted in the *AMAE* for thresholding event features.

TABLE III
ABLATION STUDY ON **MOTION CONSISTENCY** EVALUATION.

Method	N-M	N-Cal	N-C	CIF10
$\mathcal{T}_p$	0.991	0.819	0.923	0.728
MAE	**0.994**	**0.858**	0.949	**0.761**
AMAE	0.993	0.851	**0.955**	0.753

2) Ablation study on motion inconsistency evaluation: Table IV presents that the first two methods lose more accuracy than *AMAE*. Interestingly, the second method, which demonstrates the best performance in the motion consistency evaluation, achieves lower accuracy than the baseline in different motion environments. It is because the parameters in the second method that must be manually adjusted to achieve the best results; these inflexible parameters can substantially reduce the robustness of the model. Conversely, the *AMAE* benefits from our learning strategy, as neural networks can compute parameters according to different inputs to ensure the fidelity of an encoder to the semantic information. Thus, the *AMAE* is more robust to different motion information compared with the other two encoders. At the same time, these results also reflect the limitations of the first evaluation method.

TABLE IV
ABLATION STUDY ON **MOTION INCONSISTENCY** EVALUATION.

Method	N-M	N-Cal	N-C	CIF10
$\mathcal{T}_p$	0.981	0.798	0.901	0.697
MAE	0.975	0.813	0.897	0.688
AMAE	**0.984**	**0.828**	**0.942**	**0.733**

V. LATENCY AND COMPUTATIONAL TIME

Latency and computational time are central event-based classification method properties aside from accuracy and motion robustness. Various previous works have used handcraft descriptors or biological networks ($e.g.$, SNNs) to perform the classification task. However, meanwhile, these approaches achieve low latency in processing events; they sacrifice computational efficiency because each event needs to go through the entire model. By contrast, frame-based approaches sacrifice their latency in processing events to save computational time and increase their final performance. For classification tasks, the advantage of extracting detailed motion information asynchronously has become a hindrance for a classifier to achieve correct prediction. This is because a single event only reports the value of a moving point; few events carry only local-level messages about the scene. Thus it is easy to mislead the classifier's judgment ($e.g.$, cat and dog feet are similar). Based on this fact, we choose a frame-based method as the first stage in processing event signals. In Table V, we mainly compare the processing speed of different methods follow the evaluation protocol used in [12], [16] to compare our method, *AMAE*, with other methods shown in Table I on N-Cars dataset. We implement methods on a computer equipped with a CPU (Intel i7), a GPU (GeForce RTX 2080 Ti), and 16GB of RAM. The comparison results of computational time are shown in Table V, where $Time$ and kEv/s represent the average computational time per sample in N-Cars and the number of events processed per second, respectively. Although not on the order of the event rate, the frame-rate (a full forward pass only takes on the order of 5.83ms, which translates to a maximum rate of 172 Hz) is high enough for most high-speed applications.

TABLE V
COMPUTATIONAL TIME FOR 100 MS OF EVENT DATA AND NUMBER OF EVENTS PROCESSED PER SECOND.

Method	Asynchronous	Time(ms)	kEv/s
Gabor-SNN [12]	Yes	285.95	14.15
HOTS [11]	Yes	157.57	25.68
HATS [12]	Yes	7.28	555.74
Lifetime frame[5]	Yes	491.54	8.07
Synchrony frame[29]	Yes	221.59	17.90
Voxel Grid[15]	No	**5.54**	**714.5**
EST [16]	No	6.26	632.9
AMAE(Ours)	No	**5.83**	**680.4**

VI. CONCLUSIONS

In this work, we introduce the adaptive motion-agnostic encoder (*AMAE*) for object classification. Our encoder can extract indistinguishable semantic information behind event signals under various motion conditions by encoding events with integration block ($\mathcal{T}_p$) and adaptive motion-agnostic filter (*AMAF*). Moreover, the differentiable and continuous properties of the encoder allow us to learn its parameters with CNN's blocks, which makes our encoder more robust than traditional handcraft descriptors. The performance of our proposed method in motion consistency and inconsistency evaluations shows that our approach outperforms state-of-the-art approaches in terms of accuracy and motion robust-

ness. In the future, we plan to extend our model to process events asynchronously to dig more potential from the event-based data. Also, due to the lack of diversity in existing datasets, we hope to create more realistic and challenging datasets to evaluate models' motion robustness in practical applications comprehensively.

REFERENCES

[1] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, Feb 2008.

[2] C. Posch, D. Matolin, and R. Wohlgenannt, "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*, vol. 46, no. 1, pp. 259–275, 2010.

[3] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis, and D. Scaramuzza, "Event-based vision: A survey," *CoRR*, vol. abs/1904.08405, 2019. [Online]. Available: http://arxiv.org/abs/1904.08405

[4] R. Benosman, C. Clercq, X. Lagorce, S.-H. Ieng, and C. Bartolozzi, "Event-based visual flow," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 2, pp. 407–417, 2013.

[5] E. Mueggler, C. Forster, N. Baumli, G. Gallego, and D. Scaramuzza, "Lifetime estimation of events from dynamic vision sensors," in *2015 IEEE international conference on Robotics and Automation*. IEEE, 2015, pp. 4874–4881.

[6] D. Weikersdorfer and J. Conradt, "Event-based particle filtering for robot self-localization," in *2012 IEEE International Conference on Robotics and Biomimetics*. IEEE, 2012, pp. 866–870.

[7] A. Censi and D. Scaramuzza, "Low-latency event-based visual odometry," in *2014 IEEE International Conference on Robotics and Automation*. IEEE, 2014, pp. 703–710.

[8] H. Kim, A. Handa, R. Benosman, S.-H. Ieng, and A. Davison, "Simultaneous mosaicing and tracking with an event camera," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014.

[9] G. Gallego, J. E. A. Lund, E. Mueggler, H. Rebecq, T. Delbruck, and D. Scaramuzza, "Event-based, 6-dof camera tracking from photometric depth maps," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2402–2412, Oct 2018.

[10] H. Kim, S. Leutenegger, and A. J. Davison, "Real-time 3d reconstruction and 6-dof tracking with an event camera," in *European Conference on Computer Vision*. Springer, 2016, pp. 349–364.

[11] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman, "Hots: a hierarchy of event-based time-surfaces for pattern recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 7, pp. 1346–1359, 2016.

[12] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman, "Hats: Histograms of averaged time surfaces for robust event-based object classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1731–1740.

[13] A. I. Maqueda, A. Loquercio, G. Gallego, N. García, and D. Scaramuzza, "Event-based vision meets deep learning on steering prediction for self-driving cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5419–5427.

[14] A. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Ev-flownet: Self-supervised optical flow estimation for event-based cameras," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.

[15] A. Zihao Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based optical flow using motion compensation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 0–0.

[16] D. Gehrig, A. Loquercio, K. G. Derpanis, and D. Scaramuzza, "End-to-end learning of representations for asynchronous event-based data," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5633–5643.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] W. Maass, "Networks of spiking neurons: the third generation of neural network models," *Neural networks*, vol. 10, no. 9, pp. 1659–1671, 1997.

[20] G. Orchard, C. Meyer, R. Etienne-Cummings, C. Posch, N. Thakor, and R. Benosman, "Hfirst: a temporal approach to object recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 10, pp. 2028–2040, 2015.

[21] B. Zhao, R. Ding, S. Chen, B. Linares-Barranco, and H. Tang, "Feedforward categorization on aer motion events using cortex-like features in a spiking neural network," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 9, pp. 1963–1978, 2014.

[22] A. Amir, B. Taba, D. Berg, T. Melano, J. McKinstry, C. Di Nolfo, T. Nayak, A. Andreopoulos, G. Garreau, M. Mendoza, *et al.*, "A low power, fully event-based gesture recognition system," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7243–7252.

[23] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased lstm: Accelerating recurrent network training for long or event-based sequences," in *Advances in neural information processing systems*, 2016, pp. 3882–3890.

[24] X. Clady, S.-H. Ieng, and R. Benosman, "Asynchronous event-based corner detection and matching," *Neural Networks*, vol. 66, pp. 91–106, 2015.

[25] E. Mueggler, C. Bartolozzi, and D. Scaramuzza, "Fast event-based corner detection." in *British Machine Vision Conference*, 2017.

[26] V. Vasco, A. Glover, and C. Bartolozzi, "Fast event-based harris corner detection exploiting the advantages of event-driven cameras," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2016, pp. 4144–4149.

[27] S. Seifozzakerini, W.-Y. Yau, B. Zhao, and K. Mao, "Event-based hough transform in a spiking neural network for multiple line detection and tracking using a dynamic vision sensor." in *British Machine Vision Conference*, 2016.

[28] E. Mueggler, H. Rebecq, G. Gallego, T. Delbruck, and D. Scaramuzza, "The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam," *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 142–149, 2017.

[29] X. Clady, J.-M. Maro, S. Barré, and R. B. Benosman, "A motion-based feature for event-based pattern recognition," *Frontiers in neuroscience*, vol. 10, p. 594, 2017.

[30] T. Brosch, S. Tschechne, and H. Neumann, "On event-based optical flow detection," *Frontiers in neuroscience*, vol. 9, p. 137, 2015.

[31] G. Orchard and R. Etienne-Cummings, "Bioinspired visual motion estimation," *Proceedings of the IEEE*, vol. 102, no. 10, pp. 1520–1536, 2014.

[32] G. Gallego, H. Rebecq, and D. Scaramuzza, "A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3867–3876.

[33] J. Xu, M. Jiang, L. Yu, W. Yang, and W. Wang, "Robust motion compensation for event cameras with smooth constraint," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 604–614, 2020.

[34] T. Stoffregen, G. Gallego, T. Drummond, L. Kleeman, and D. Scaramuzza, "Event-based motion segmentation by motion compensation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7244–7253.

[35] A. Mitrokhin, C. Fermüller, C. Parameshwara, and Y. Aloimonos, "Event-based moving object detection and tracking," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2018, pp. 1–9.

[36] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning*, 2010, pp. 807–814.

[37] G. Orchard, A. Jayawant, G. K. Cohen, and N. Thakor, "Converting static image datasets to spiking neuromorphic datasets using saccades," *Frontiers in neuroscience*, vol. 9, p. 437, 2015.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.