Supplemental Material for the Paper
"HATS: Histograms of Averaged Time Surfaces for Robust Event-based Object Classification"

Amos Sironi[1] , Manuele Brambilla[1], Nicolas Bourdis[1], Xavier Lagorce[1], Ryad Benosman[1,2,3]
[1]PROPHESEE, Paris, France [2]Institut de la Vision, UPMC, Paris, France
[3]University of Pittsburgh Medical Center / Carnegie Mellon University
{asironi, mbrambilla, nbourdis, xlagorce}@prophesee.ai ryad.benosman@upmc.fr

In this appendix, we report additional results and details which did not fit in our submission due to space limitations. In Section A, we present additional details about the N-CARS dataset introduced in Section 5 of the paper. Then, in Section B, we provide the parameters we used for the baseline methods presented in Section 6. Finally, in Section C, we compare the ROC curves for the *HATS*, *HOTS* and *Gabor-SNN* methods on the N-CARS dataset.

A. N-CARS Dataset: Additional Details

In this section, we provide additional information on how the N-CARS dataset was created and how the ground truth annotations were generated. The N-CARS dataset is available for download at `http://www.prophesee.ai/dataset-n-cars/` A video containing some samples from the datset is also provided together with the supplementary material (`car_examples.avi`, `background_examples.avi`).

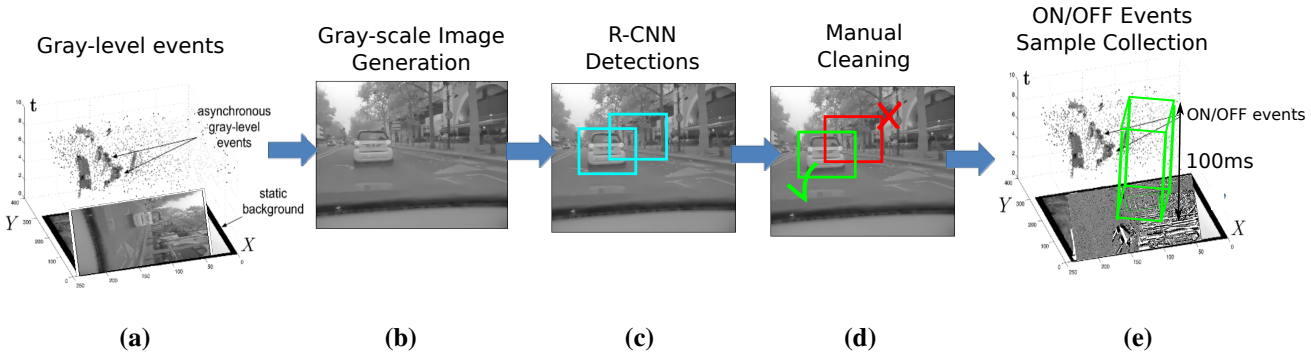

Figure 1: Semi-automated labeling protocol based on a single ATIS camera. **(a)** Asynchronous gray-scale events are generated by the ATIS camera. **(b)** By cumulating gray-scale events, we create gray-scale images every 100ms. **(c)** We run state-of-the-art R-CNNs [6, 7] to automatically detect bounding-boxes containing cars or background samples. **(d)** The bounding-boxes are manually cleaned to ensure the dataset contains only correctly labeled samples. **(e)** The spatial-bounding box is converted to a spatio-temporal bounding-box of 100ms containing ON/OFF events. Each event stream contained in the spatio-temporal bounding-box corresponds to a sample in the N-CARS dataset. This figure is best seen in color.

The raw data from which the dataset is generated, were collected in the following way. We mounted an ATIS camera (Section 2) behind the windshield of a car and recorded different driving sessions. Approximately 80 minutes of video were recorded over multiple sessions in urban environments and on a highway. The videos contain ON/OFF *events* (in the form of x-y position in the pixel array, timestamp, and polarity) and the measure of the absolute luminous intensity at the corresponding pixel.

We extracted standard gray-scale images by cumulating the absolute luminous intensity every 100ms (Fig. 1(a,b)). Each gray-scale image was tagged with a timestamp equal to the time at which the cumulating process was completed, that is, the first image has timestamp 100, the second 200, the third 300, and so on.

Form the recordings, we generate a dataset for a car/background binary classification task, using the following semi-automatic labeling protocol. First, we process the gray-scale images using state-of-the-art object detectors [6, 7] (Fig. 1(c)). For every gray-scale image, the detectors return a list of bounding-boxes in the form of x-y position, width and height dimensions, class (that is, whether the example contains a car or background) and a measure of the confidence between 0 and 1 of this classification. To each bounding-box, we associate the timestamp of the image it was extracted from.

Subsequently, we discard bounding-boxes with too low confidence score, keeping only those with confidence greater than 0.5 for car examples and 0.9 for background examples. We also discard bounding-box with dimensions not respecting the following conditions: width between 30 and 120 pixels; height between 25 and 100 pixels; ratio between width and height between 0.5 and 2. Finally, each remaining bounding-box was manually checked to verify that it was correctly positioned around a car (in case of car examples), or to check that it did not contain any car (in case of background examples) Fig. 1(d).

The list of manually cleaned bounding-boxes in the gray-scale images was then used to create an event-based dataset from the ON/OFF events. A sample in the event-based dataset is composed of all events with x-y position inside the spatial coordinates of the corresponding bounding-box and with timestamp in $[t-100ms, t]$ where t is the timestamp of the bounding-box. The class of the sample is the same as the class of the bounding-box (Fig. 1(c)). To avoid including in the dataset samples without enough events, we discard all samples containing less than 500 events.

Finally, the sequences of events where spatially and temporally shifted so that the top-leftmost event is at the origin (x and y coordinates equal to 0) and so that the first event has a timestamp equal to zero. In Table 1, we report the total number of samples in the N-CARS dataset. In Fig. 2, we report the histogram of the number of events contained in the N-CARS dataset (in logarithmic scale).

Table 1: Number of samples in the N-CARS dataset.

N-Cars	Car examples	Background examples	**Total**
Training set	7940	7482	15422
Testing set	4396	4211	8607
Total	12336	11693	**24029**

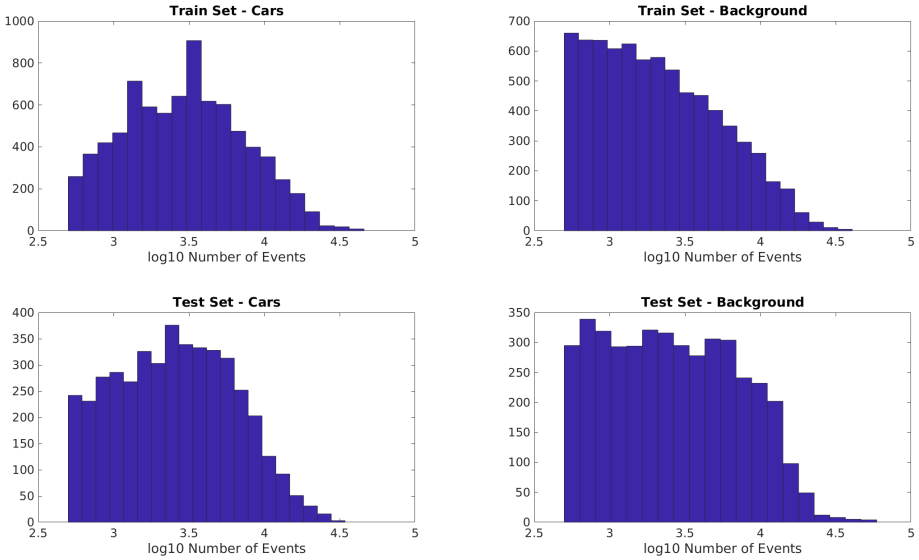

Figure 2: Logarithmic histogram of the number of events per sample in the N-CARS dataset.

B. Baseline Methods: Implementation Details

In this section, we first give more details about the datasets converted from frames of Sec. 5.1 of the paper, then present the parameters used for the baseline methods used in Section 6 of the paper.

As the original frame-based dataset, N-MNIST consists of 10 different classes, with a total of 60,000 training and 10,000 test samples of handwritten digits. The samples have size of 35x35 pixels and a duration of approximately $300ms$.

MNIST-DVS instead, is composed of a subset of the original MNIST dataset and contains 10,000 samples, generated at three different resolutions. In our experiments, we used scale 4 and used 90% of the samples for training and 10% for testing, as usually done for this dataset. Training and evaluation were repeated 10 times with different random splits. The resolution of the samples is 128×128 pixels and the duration of a presentation around $2.3s$.

N-Caltech101 consists of 100 different object classes and a background class. Each category has between 31 and 800 images, for a total of 8,242 images. The resolution of each sample is approximately 300x200 pixels and the duration is approximately $300ms$. In our experiments, we use two thirds of the samples of each class for training and the rest for testing.

Finally, CIFAR10-DVS contains 10 classes each with 1,000 presentations, for a total of 10,000 samples of resolution 128×128 and duration of about $1.2s$. We used the same training protocol as for the MNIST-DVS dataset.

The parameters for *H-First*, *HOTS*, *Gabor-SNN* and *HATS* on the different datasets are given in Tab. 2, Tab. 3, Tab. 4 and Tab. 5, respectively. For the *H-First* and *HOTS*, we used the same notation of as in [5, 4] and more details on the meaning of the parameters can be found in [5, 4].

For *Gabor-SNN* , we used a spiking neural network composed of 2 layers of *Gabor filters*. A Gabor filter [1] is a linear filter commonly used in Computer Vision and it is also often used as a predefined filter for the first layers of frame-based Convolutional Neural Networks [2]. A Gabor filter is defined as:

$$g(x, y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi\frac{x'}{\lambda} + \psi\right)\right), \tag{1}$$

where $x' = x\cos\theta + y\sin\theta$ and $y' = -x\sin\theta + y\cos\theta$, λ is the wavelength of the sinusoidal component, σ is the standard deviation of the Gaussian envelope, θ represents the orientation of the normal to the Gabor filter, γ is the spatial aspect ratio, and ψ is the phase offset.

The first layer of the network contains two filter banks, one per polarity. Each filter bank is composed of Gabor filters at 8 different orientations (i.e. 8 different values of θ). and at k scales, for a total of $2 \times 8 \times k$ filters. For each scale k, the Gabor filters are defined by the kernel size s, and the parameters of Eq.(1). A 2×2 max pooling was applied after the first layer.

The second layer contains a Gabor filter bank per output of the first layer. The k, s, λ and σ parameters for the different datasets and the different layers are given in Tab. 4, while we used $\gamma = 0.3$ and $\psi = 0$ in all our experiments.

Table 2: *H-First* parameters used in the experiments of Section 6. For N-MNIST we report previously published results [5].

	V_{thresh}	I_l/C_m	t_{refr}	Kernel Size
N-Caltech101	100	10	5	13
MNIST-DVS	150	25	5	7
CIFAR10-DVS	100	10	5	13
N-CARS	30	25	5	13

Table 3: *HOTS* parameters used in the experiments of Section 6.

	R_1	$\tau_1(ms)$	N_1	K_R	K_τ	K_N
N-MNIST	2	35	4	2	3	2
N-Caltech101	4	35	8	2	3	2
MNIST-DVS	2	50	4	2	3	2
CIFAR10-DVS	2	40	4	2	5	2
N-CARS	2	11	4	2	3	2

Table 4: *Gabor-SNN* parameters used in the experiments of Section 6.

	k (layer 1)	s (layer 1)	λ (layer 1)	σ(layer 1)	k (layer 2)	s (layer 2)	λ (layer 2)	σ(layer 2)
N-MNIST	3	1,2,3	3.5	2.8	2	3,5	3.5	2.8
N-Caltech101	3	3,7,15	3.5	2.8	2	3,7	3.5	2.8
MNIST-DVS	3	1,2,3	3.5	2.8	2	3,5	3.5	2.8
CIFAR10-DVS	2	7,15	5	5	1	3	2.5	5
N-CARS	2	7,15	5	5	1	3	2.5	5

Table 5: *HATS* parameters used in the experiments of Section 6.

	K	ρ	$\tau(\mu s)$	$\Delta t(ms)$
N-MNIST	5	2	10^{11}	100
N-Caltech101	10	3	10^{6}	100
MNIST-DVS	5	2	10^{11}	200
CIFAR10-DVS	12	4	10^{4}	200
N-CARS	10	3	10^{9}	100

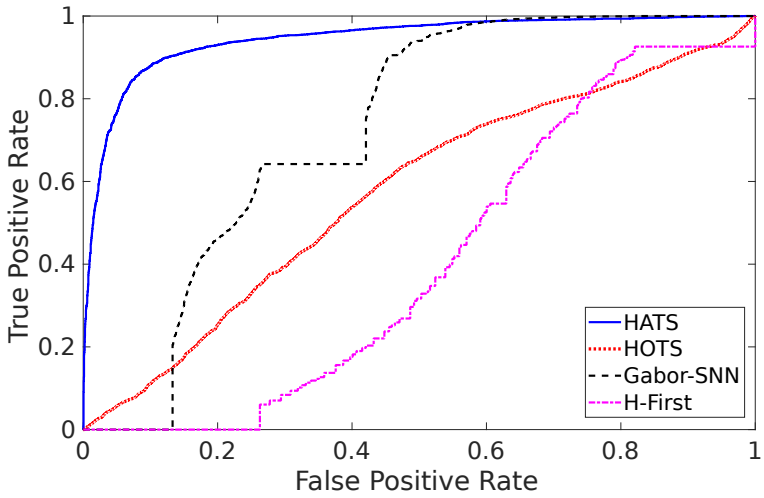

Figure 3: The ROC curve of *HATS*, *HOTS*, *Gabor-SNN* and *H-First* for the N-CARS dataset. This figure is best seen in color.

C. Additional Results on the N-CARS Dataset

In this section we present additional results on the N-CARS dataset that did not fit in the main article due to space limitations.

In the first experiment, we study the impact of the Local Memory formulation of Eq.(3) of the main article, compared to the time surface of Eq.(2) originally proposed in [4]. More precisely, we compute the histograms of Eq.(5) using Eq.(2) instead of Eq.(3). The performance on the N-CARS dataset drops from 90.2% to 81.3%, meaning that the regularization brought by the local memory time surface of Eq.(3) brings better accuracy.

In the second experiment, we compute the receiver operating characteristic (ROC) curves [3] for the N-CARS dataset for our method and the baseline methods.

The results are shown in In Figure 3. Each curve shows different values of true positive rate against false positive rate for various thresholds on the classification score. As we can see, the ROC curve for *HATS* is consistently above the curves of the other approaches, confirming the superiority of our approach.

References

[1] A. C. Bovik, M. Clark, and W. S. Geisler. Multichannel texture analysis using localized spatial filters. *TPAMI*, 1990.

[2] J. Bruna and S. Mallat. Invariant scattering convolution networks. *TPAMI*, 2013.

[3] T. Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 2006.

[4] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman. Hots: a hierarchy of event-based time-surfaces for pattern recognition. *TPAMI*, 2017.

[5] G. Orchard, C. Meyer, R. Etienne-Cummings, C. Posch, N. Thakor, and R. Benosman. Hfirst: A temporal approach to object recognition. *TPAMI*, 2015.

[6] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. *CoRR*, 2016.

[7] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.