Pseudo-labels for Supervised Learning on Dynamic Vision Sensor Data, Applied to Object Detection under Ego-motion

Nicholas F. Y. Chen
DSO National Laboratories
12 Science Park Drive, Singapore (118225)
cfangyew@dso.org.sg

Abstract

In recent years, dynamic vision sensors (DVS), also known as event-based cameras or neuromorphic sensors, have seen increased use due to various advantages over conventional frame-based cameras. Using principles inspired by the retina, its high temporal resolution overcomes motion blurring, its high dynamic range overcomes extreme illumination conditions and its low power consumption makes it ideal for embedded systems on platforms such as drones and self-driving cars. However, event-based data sets are scarce and labels are even rarer for tasks such as object detection. We transferred discriminative knowledge from a state-of-the-art frame-based convolutional neural network (CNN) to the event-based modality via intermediate pseudo-labels, which are used as targets for supervised learning. We show, for the first time, event-based car detection under ego-motion in a real environment at 100 frames per second with a test average precision of 40.3% relative to our annotated ground truth. The event-based car detector handles motion blur and poor illumination conditions despite not explicitly trained to do so, and even complements frame-based CNN detectors, suggesting that it has learnt generalized visual representations.

1. Introduction

Dynamic vision sensors (DVS), also known as event-based cameras or neuromorphic sensors [3, 14], are a class of biologically-inspired sensors which capture data in an asynchronous manner. When a pixel detects a change in luminance above a certain threshold in log scale, the device emits an output (an 'event') containing the pixel location, time and polarity (+1 or -1, corresponding to an increase or decrease in luminance respectively). Such sensors have a temporal resolution on the order of milliseconds or less, making the device suitable for high speed recognition, tracking and collision avoidance. Other advantages of dynamic vision sensors include a high dynamic range and power efficiency, making it ideal for outdoor usage on embedded systems in robotics.

Frame-based labeled data sets are widely available, contributing to the tremendous advancements in frame-based computer vision in recent years. However, event-based computer vision is still in the process of maturing, and current event-based data sets are quite limited, especially in the case of object detection. Event-based data sets have been released for robotics applications such as simultaneous localization and mapping (SLAM), visual navigation, pose estimation and optical flow estimation [2, 17, 27, 30], and comprise of mostly indoor scenes with simple objects and occasional outdoor scenes. For object recognition and detection, some data sets were created by placing a dynamic vision sensor in front of a monitor and recording existing frame-based data sets [10, 20]. Moeys *et al*. [16] recorded scenes of a predator robot chasing a prey robot in a controlled lab environment with some background objects.

In the long run, dynamic vision sensors might be integrated in platforms such as drones and autonomous vehicles which work in complex, outdoor environments. The DAVIS Driving Dataset 2017 (DDD17) [4] is the largest data set as of writing which captures such environments, with over 400 GB and 12 hours worth of driving data spread across over 40 scenes at a resolution of 346 × 260 pixels. These scenes are varied over the times of the day (day, evening, night), weather (dry, rainy, wet) and location (campus, city, town, freeway, highway), and includes vehicle details like velocity, steering wheel angle and accelerator pedal position. The DAVIS is a camera model which contains a dynamic vision sensor synchronized with a grayscale frame-based camera (also known as the active pixel sensor, or APS).

High speed object detection under ego-motion from dynamic vision sensor data serves a few purposes. First, dynamic vision sensors overcome problems which

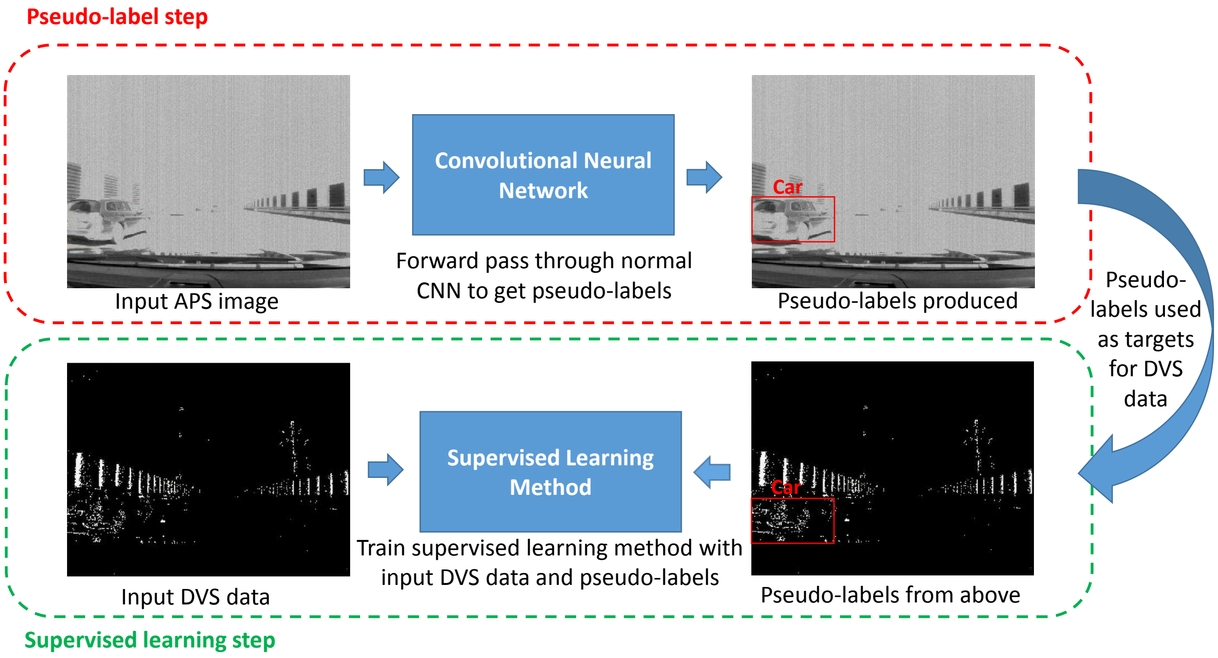
Pseudo-label step
Input APS image
Convolutional Neural Network
Forward pass through normal CNN to get pseudo-labels
Car
Pseudo-labels produced
Pseudo-labels used as targets for DVS data
Input DVS data
Supervised Learning Method
Train supervised learning method with input DVS data and pseudo-labels
Car
Pseudo-labels from above
Supervised learning step

Figure 1. Schematic of the proposed pseudo-labeling and supervised learning method. Top frame: Image from the APS sensor is passed through a CNN to get the pseudo-labels. Bottom frame: Since the dynamic vision sensor is synchronized with the APS (grayscale) camera, the pseudo-labels from the previous step is treated as ground truth to train a supervised learning method which takes dynamic vision sensor data as inputs.

ordinary frame-based cameras typically encounter. At high speeds, frame-based cameras suffer from motion blur and collision avoidance is limited, placing a speed limit on the platform which the camera is mounted on. In extreme illumination conditions, frame-based cameras have difficulty capturing features of objects. Since dynamic vision sensors output changes in luminance, the data is a sparse representation which can be processed faster, compared to the output of frame-based cameras which contains (potentially redundant) background information. Also, detections from dynamic vision sensor data can be used to complement detections from frame-based cameras, as we will show from our experiments. Last, detection under ego-motion is required because dynamic vision sensors mounted on platforms will inevitably have ego-motion, and the output of the sensors will include some background information as a result, creating distractions which the detection algorithm must overcome.

Like most objects in event-based data sets however, objects in the DDD17 are not labeled. In this paper, we take advantage of the mature state of frame-based detection by using a state-of-the-art CNN to perform car detection on the grayscale (APS) images of the DDD17. These detections, hence termed ‘pseudo-labels’, are shown to be effective when used as targets for a separate (fast) CNN when training on dynamic vision sensor data in the form of binned frames. A schematic of this method can be found in Figure 1.

Contributions

1. We trained a CNN on pseudo-labels to detect cars from dynamic vision sensor data, with a test average precision of 40.3% relative to annotated ground truth. This is the first time that high-speed (100 FPS) object detection is done on dynamic vision sensor data under ego-motion in a real environment, whereas previous works have only focused on recognizing/detecting simple objects in a controlled environment or detecting objects without camera ego-motion.

2. We show that a CNN trained on pseudo-labels can detect cars despite motion blur or poor lighting, even though pseudo-labels were not generated for these scenarios. This CNN even complements the original frame-based CNN that was used to generate the pseudo-labels, suggesting that our trained CNN learnt generalized visual representations of cars.

1.1. Related work

Pseudo-labels & cross modal distillation Pseudo-labeling was introduced by Lee [12] for semi-supervised learning on frame-based data, where during each weight update, the unlabeled data picks up the class which has the maximum predicted probability and treats it as the ground truth. Pathak *et al*. [22] used automatically generated masks (pseudo-labels in their context) from unsupervised motion segmentation on videos, and then trained a CNN to predict these masks from static images. The trained CNN learnt feature representations and was able to perform image classification, semantic segmentation and object detection.

For data sets with paired modalities (e.g. RGB-D data contains RGB data of a scene synchronized with depth data of the same scene), cross modal distillation [8] is a scheme that transfers knowledge from one modality, which has a lot of labels, to another modality, which has very few labels. In [8], mid-level representations of a CNN trained on RGB images were used to supervise training for another CNN to perform object detection and segmentation on depth images. In [1], the visual modality of videos was used to generate pseudo-labels from CNNs and used to train a separate 1-D CNN to classify scenes from sound inputs. Our work is inspired by these cross modal methods, and we leverage on the fact that the DDD17 is a large data set with synchronized DVS and APS modalities.

Event-based object detection Object detection on dynamic vision sensor data is relatively new since labeled event-based data sets are scarce. Liu *et al*. [15] performed object detection on the predator-prey data set [16]. They used dynamic vision sensor data as an attention mechanism for a frame-based CNN, and compared it to using a CNN to perform detection on the entire grayscale image. Including particle filter for both methods to aid tracking, the former method is 70X faster than the latter, with an accuracy of 90%. Li *et al*. [13] proposed a method which adaptively pools feature maps from successive frames (generated by binning dynamic vision sensor data over time) to create motion invariant features for object detection. They demonstrated hand detection with performance scores averaging from 61.3% to 76.0% depending on the variant of the method used. Hinz *et al*. [9] demonstrated a tracking-by-clustering system which detects and tracks vehicles on a highway bridge. Both [13] and [9] did not benchmark their methods on dynamic vision sensor data under camera ego-motion.

2. Generating pseudo-labels for dynamic vision sensor data

We overcome the lack of labeled dynamic vision sensor data by using cross modal distillation with pseudo-labels on the DDD17 data set (see Figure 1 for a brief outline). Since the DAVIS sensor has a frame-based camera (APS) synchronized with a dynamic vision sensor, the ground truth in one camera is the same as the ground truth in the other camera. The grayscale (APS) images are fed into a state-of-the-art CNN which generates outputs (pseudo-labels). These pseudo-labels with confidence above a threshold are treated as ground truth and used to train a supervised learning method, which takes the dynamic vision sensor data as inputs. Though the pseudo-labels are noisy, Pathak *et al*. [22] argues that in the absence of systematic errors, such noise are perturbations around the ground truth, and since supervised learning methods like neural networks have a finite capacity, it cannot learn the noise perfectly and it might learn something closer to the ground truth. In the context of our experiments (car detection), the pseudo-labels are bounding boxes while the supervised learning method is also a CNN. Pseudo-labeling is not limited to object detection–it should work for other computer vision tasks like image segmentation, image recognition and activity recognition.

Implementation Details We chose the Recurrent Rolling Convolution (RRC) [25] CNN as the object detection CNN for APS images because as of writing, it is the best-performing model on the KITTI Object Detection Evaluation benchmark [6]. The APS images are in grayscale and due to domain shift, the original RRC trained on the KITTI data set (which is in RGB) might not produce high quality pseudo-labels. As such, we also use another model of the RRC which is fine-tuned over 1000 iterations on a grayscale-converted KITTI data set. This will allow us to investigate how pseudo-labels of differing quality affect the performance of the trained DVS detector. As the RRC takes in images of a different aspect ratio than the APS images, we scaled the APS images to the largest possible size while preserving the aspect ratio, and padded the remainder of the image with zeroes. By keeping predictions that have at least a 0.5 confidence score, we produced about 330k and 400k pseudo-labeled images from the original and fine-tuned RRC respectively for various day and evening scenes (the RRC might not produce accurate detections for the night scenes). The scenes are split into train/val/test sets in the ratio 71/15/14 by their recording length, with each set covering a variety of conditions and scenes (more details in the supplementary material). We focused only on detecting cars, but this method can easily be extended to other classes.

3. Supervised learning with pseudo-labels

Implementation Details We adopt a frame-based approach to the dynamic vision sensor data for object detection, because frame-based object detection is mature. The dynamic vision sensor data are converted to images by binning the dynamic vision sensor outputs in 10 ms intervals, and each pixel takes the value

$$\sigma(x) = 255 * \frac{1}{1 + e^{-x/2}}, \qquad (1)$$

where x is the sum of the polarities of the events in the 10 ms interval. We refer to this as the *sigmoid representation* of the dynamic vision sensor data, chosen because it is a simple way to ensure that all values will lie in [0, 255]. While there exist various image representations of DVS data [7, 16] which could affect the final performance of our detector, we leave such optimization to future work. 10 ms bin size was chosen for 2 reasons: (i) We aim to achieve detection at 100 frames per second (FPS), about an order of magnitude above most state-of-the-art CNNs. (ii) Ideally we would set the bin size to be as small as possible to reduce the effects of motion blur, but this is limited by the accompanying method which must handle the data at the right frame rate. 10 ms is a reasonable bin size given these considerations.

We used the tiny YOLO architecture [23, 24] for the DVS detector because it was shown to run at 207 FPS on a Geforce GTX Titan X GPU (Maxwell) with a decent performance of 57.1% mean average precision on the VOC 2007+2012 benchmark. It requires about 7 billion multiply and accumulate operations, which is at least 100 times less than that of the RRC. We started with this CNN pre-trained on the VOC 2007+2012 benchmark and fine-tuned it using the pseudo-labels generated, in steps of 10k iterations, up to 150k iterations (including the 20k iterations from pre-training). As we want to show that the object detection CNN performs well as a result of the effectiveness of pseudo-labels rather than the result of optimizing hyper-parameters, we only changed the subdivisions from 8 to 4 and batch size from 64 to 128, and kept the other settings as provided in [23, 24].

3.1. Quantitative results

The scenario that we are tackling (high-speed object detection in a real environment from dynamic vision sensor data under camera ego-motion) is the first of its kind, so there are no other state-of-the-art algorithms for comparison. As such, we hope that this work serves as a benchmark for future methods tackling the same scenario.

Since there is no ground truth data for the objects in DDD17, we measure performance relative to the RRC pseudo-labels during the model validation step. The model with the highest average precision on the validation set will then be evaluated on the test set. We use an intersection-over-union (IoU) threshold of 0.5 for this step.

Evaluation against ground truth We randomly selected 1000 frames from the test set for manual annotation, and all performance figures reported henceforth are obtained by evaluation on this subset. Similar to the KITTI object detection benchmark, we only consider objects that have a minimum height of 25 pixels. A summary of the results can be found in table 1 and the corresponding precision-recall curves can be found in Figure 2. The test average precision of the DVS-only detector is 36.9% and 40.3% for pseudo-labels generated by the RRC (original) and the RRC (fine-tuned) respectively, at an IoU threshold of 0.5. This effect will be explained later. As a comparison, the tiny YOLO architecture achieves a mean average precision of 57.1% when trained on real labels (VOC 2007+2012 benchmark). Note that the RRC's performance on our test set is much lower than that reported for the KITTI data set (87.4% for the hard setting, IoU threshold at 0.7) because the data set we are using has a lower resolution (346 × 260 vs 2560 × 768 for KITTI).

Modality	Arch.	AP@0.5	AP@0.7
APS	RRC	44.1%	39.6%
DVS	t.YOLO	36.9%	18.3%
APS+DVS	RRC+t.YOLO	55.6%	39.9%
APS	RRC(ft)	53.7%	47.2%
DVS	t.YOLO(ft)	**40.3**%	**19.9**%
APS+DVS	RRC+t.YOLO(ft)	62.2%	47.7%

Table 1. Evaluation results of our experiments, at IoU thresholds of 0.5 and 0.7. Keys: Arch.=Architecture, ft=fine-tune, AP=average precision, t.YOLO(ft)= tiny YOLO model trained on pseudo-labels produced by RRC (fine-tuned). We can see that the DVS-only detector has higher performance when trained on pseudo-labels by RRC (fine-tuned) compared to that of RRC (original). Also, the combination of the APS (grayscale) and DVS cameras help achieve a higher performance than either modality alone.

Complementing DVS and grayscale detections We evaluated if the combination of DVS and grayscale detections can improve the overall performance, listed as APS+DVS in table 1. We combined the detections of the DVS-only detector and the RRC, and applied non-maximum suppression with an IoU threshold of 0.4 to remove duplicates. At a detection IoU threshold of 0.5, such a combination yielded an average precision of 62.2% on our annotated ground truth data set, roughly a 16% increase over using only the RRC. This is despite the fact that the DVS-only detector is trained only on knowledge generated by the RRC, showing that the DVS-only detector has learnt generalized representations of cars. A similar effect was observed in [8] for the RGB and depth modalities.

Given the current state of hardware, the RRC is not a real-time detector and the specific combination of the detections mentioned above is not practical yet. However, we hope that this experiment will inspire future work on using detections from the DVS to complement detections from the APS.

We notice that at an IoU threshold of 0.7, the benefit from combining the detectors is marginal. This is because

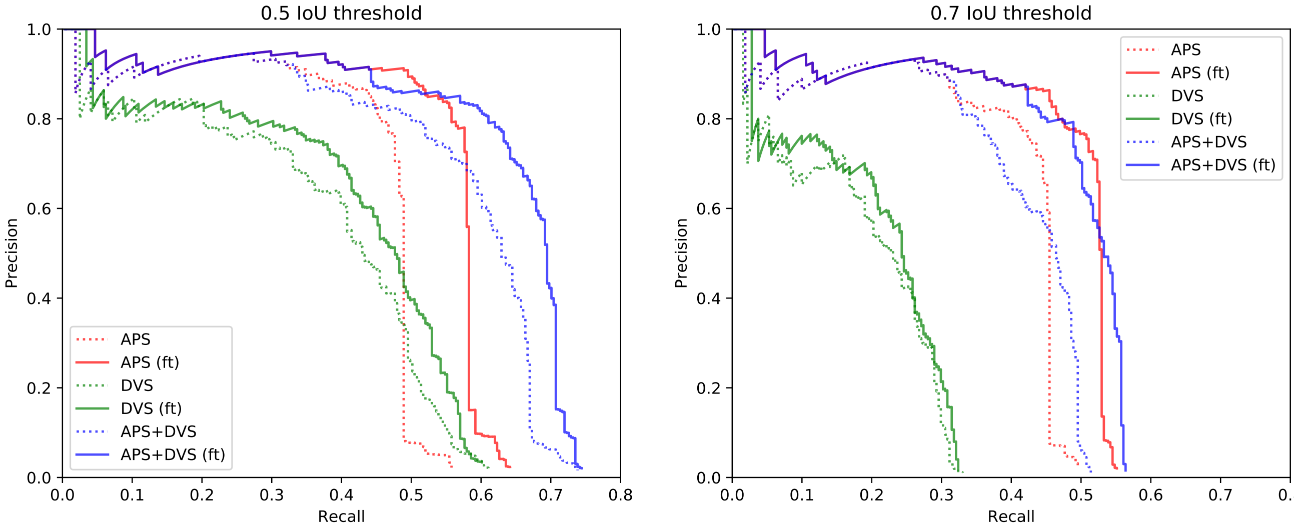

Figure 2. (Best viewed in color) Precision-recall curves for various modalities at IoU thresholds of 0.5 and 0.7.

the RRC architecture is specifically designed to work well at high IoU thresholds, whereas the tiny YOLO architecture is designed assuming that it will be evaluated at an IoU threshold of 0.5.

Comparing DVS and grayscale detections We measured the correct detections made by the detectors, regardless of the confidence score, as a fraction of the total number of ground truth objects (*i.e.* the recall) in table 2. We also take a look at the union and intersection of these detections. At 0.5 IoU threshold, the DVS-only detector picked out 60.1% of the objects while the RRC picked out 64.2% of the objects. 10.6% of the objects were detected by the DVS-only detector but not by the RRC, reinforcing the fact that the DVS-only detector learnt general representations of cars, though it was trained on the knowledge from the RRC. We notice that fine-tuning the RRC did not change the fraction by much for the DVS and APS∪DVS modalities though it improved the average precision in table 1–This might be due to the fine-tuning process increasing the confidence of correct detections rather than the number of correct detections made by the DVS-only detector.

Correctness of pseudo-labels We evaluated the correctness of pseudo-labels by taking the RRC predictions which have above 0.5 confidence (the criteria for selecting pseudo-labels), and calculated the precision and recall relative to the annotated ground truth in table 3. We observe that at least 79.9% of the pseudo-labels are correct (precision). Also, the pseudo-labels only highlight 51.4% of all ground truth objects (recall) at best, however this is not a problem since we exclude frames with no detections during training. Notice that fine-tuning actually decreases the precision but increases the recall, which means that the RRC (fine-tuned) produced more labels which cover more ground truth objects than the RRC (original), at the expense of making more false detections. Reconciling this with the results in table 1, we conclude that the increased performance of the DVS detector (trained on pseudo-labels produced by the RRC (fine-tuned)) is attributable to a larger training set, and not due to more accurate labels. This agrees with the intuition of Pathak *et al.* mentioned in the beginning of section 2 that neural networks can overcome noise from pseudo-labels.

Modality	Arch.	Frac.@0.5	Frac.@0.7
APS	RRC	55.8%	49.5%
DVS	t.YOLO	61.4%	33.0%
APS∩DVS	RRC+t.YOLO	41.7%	25.2%
APS∪DVS	RRC+t.YOLO	75.4%	57.3%
APS	RRC(ft)	64.2%	55.1%
DVS	t.YOLO(ft)	60.1%	32.4%
APS∩DVS	RRC+t.YOLO(ft)	49.5%	27.1%
APS∪DVS	RRC+t.YOLO(ft)	74.8%	60.4%

Table 2. Correct detections made by the detectors as a fraction of all the actual objects (recall), for IoU thresholds of 0.5 and 0.7. Keys: Arch.=Architecture, ft=fine-tune, Frac.=fraction, t.YOLO(ft)= tiny YOLO model trained on pseudo-labels produced by RRC (fine-tuned). Notice that the DVS detected some objects which are not detected by the RRC.

Run-time analysis The DVS detector was run on the entire test set on a GeForce GTX Titan X (Maxwell) GPU and averaged at 7 ms per image, or 142 FPS.

ft	Prec.@0.5	Rec.@0.5	Prec.@0.7	Rec.@0.7
No	88.4%	38.0%	82.6%	35.5%
Yes	87.3%	51.4%	79.9%	47.0%

Table 3. Precision and recall of RRC detections with confidence above 0.5, for IoU thresholds of 0.5 and 0.7. This gives us an idea of the correctness of pseudo-labels. Keys: ft=fine-tune, Prec.=precision, Rec.=recall. At least 79.9% of the pseudo-labels are correct (precision). Fine-tuning decreases the precision and increases the recall.

3.2. Qualitative results

Though we used the sigmoid representation for training our detector, the following images from the dynamic vision sensor are displayed in the *binary representation* for easier viewing, where each pixel in the frame takes the value of 0 if the sum of the polarities of the events in the 10 ms interval is 0, and 255 otherwise. The numbers above the bounding boxes indicate the confidence, and the threshold for displaying the bounding boxes on the following images and videos is 0.5. All bounding boxes shown are a result of the fine-tuned RRC and the DVS which is trained on its pseudo-labels. Links to videos can be found in the supplementary material, and the reader is strongly encouraged to randomly sample clips from all videos to gauge the performance of the DVS-only detector.

Daytime and evening detections Randomly sampled images from the test sets are shown in Figure 3. While the CNN is able to detect cars in the near-field, cars in the far-field and cars moving at the same velocity as the camera (hence zero relative velocity) only show up on the DVS images as thin outlines at best and as such are not detected by our CNN. This is the main drawback of the DVS.

Overcoming motion blur In the first pair of row 2 and second pair of row 6 of Figure 3, we see the high temporal resolution of the dynamic vision sensor in action. The camera is moving fast and as a result, the features captured by the frame-based camera are blurred, whereas the features captured by the dynamic vision sensor is still reasonably sharp. The cars were detected by our event-based detector but not by the RRC, reinforcing our motivation for object detection on dynamic vision sensor data. An additional motion blur scene can be found at the 1:30 mark in the video of the third test scene (see supplementary material).

Nighttime detections One key feature of dynamic vision sensors is the high dynamic range which can cope with a wide spectrum of illumination conditions. Figure 4 shows a night scene (2:01:59 mark of the night scene video in the supplementary material) where illumination is poor on the left hand side of the lane. The edges in the DVS image are visible enough that the cars are detected by the DVS-only detector, but the cars in the APS image are dark enough to blend into the surrounding and was not detected by the RRC, showing that poor illumination conditions still pose a challenge for conventional frame-based cameras. Considering that the DVS-only detector is trained only on day and evening scenes, the fact that it was able to detect cars at night shows that the detector learnt representations of cars which are robust to illumination conditions.

Limitations In Figure 5, we see an example where our approach fails. This scene is on a highway at night (see video in the supplementary material), where the light source is dominated by the headlights of the cars. As the CNN is trained on DVS images of cars in the day and evening scenes, it learns the features that are visible during those times (*e.g.* edges of the car) and it does not learn the features of the headlights. To learn such features, we require labeled data which might be hard to obtain from the pseudo-labeling method because conventional CNNs do not work well on images with poor illumination conditions. This strongly suggests that the naïve approach of binning DVS data and creating images is not sufficient to represent the data.

4. Discussion

Our implementation is largely unoptimized, and the average precision can be increased via many ways. For example, we can fine-tune the threshold to keep pseudo-labels for training, the network and learning hyper-parameters of the DVS-only detector and explore other representations of the DVS data (*e.g.* possibly binning the data by a fixed number of events). We can also combine detection with tracking methods [11, 15, 18, 29].

In Figure 3, we saw how our CNN missed detections of cars that are far away, because the pixels that spike are sparsely distributed and possibly drowned out by noise. One solution is to use a higher resolution camera to capture the features of the car with more pixels. Another solution is to move away from a frame-based approach when analyzing dynamic vision sensor data, and towards an entirely event-based approach, *i.e.* algorithms which accept sparse DVS data and take temporal information into account.

The performance of the detector, in terms of speed and average precision, is bounded by the tiny YOLO architecture. Recent research in spiking neural networks (SNNs) for image classification have shown accuracies of over 90% while operating at the millisecond time scale [19, 26, 28]. This suggests that the fully event-based approach we seek could be in the form of using the event-based ROI approach in [15] for region proposals combined with SNNs for recognition, or even an SNN for object detection. These approaches can also be trained with pseudo-labels.

Our detector requires a Titan X GPU which, in terms of power consumption, may be feasible for autonomous cars but not for drones. The power consumption of SNNs implemented on neuromorphic computing hardware is on

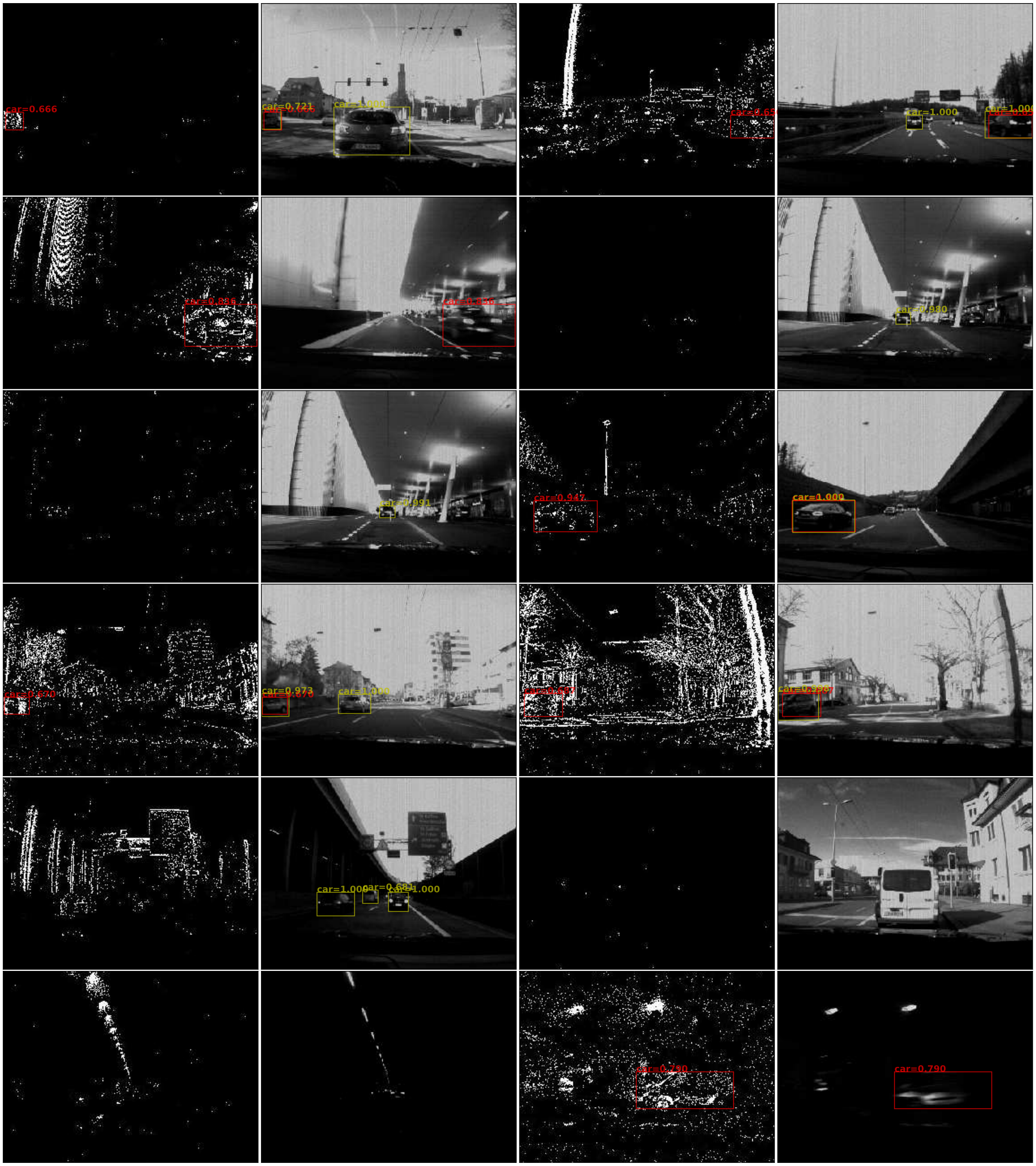

Figure 3. (Best viewed in color and zoomed in) Randomly selected images from day and evening scenes (test set). Images come in pairs: The left image of each pair is the DVS image with bounding boxes (in red) produced by DVS-only detection, while the right image of each pair is the APS image with DVS-only bounding boxes (in red) copied over and the RRC detections (in yellow) for comparison. First row, first pair: DVS fails to detect a stationary car. Second row, first pair: DVS-only detector detects a car despite motion blur, but the RRC fails to do so. Notice that in the DVS image, the edges of the car are still reasonably distinct. Second row, second pair: An example where a car in the far-field does not trigger a response in the DVS. Last row, second pair: Despite dim lighting and motion blur, the edges of the car are still visible on the DVS image and hence is detected by the DVS-only detector but not the RRC.

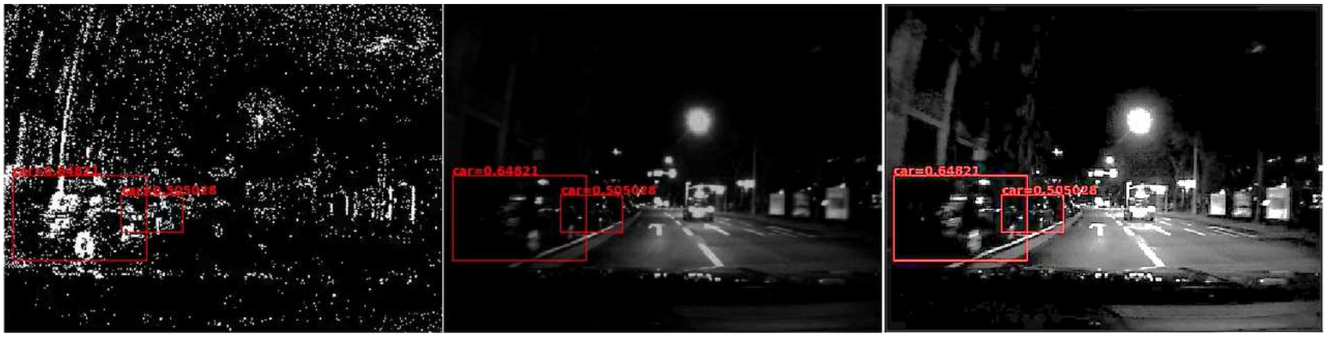

Figure 4. Left: Night scene on the DVS sensor, with bounding boxes produced by the DVS-only detector. Middle: APS image, with bounding boxes copied from DVS-only detection for comparison. Right: Middle image digitally enhanced for the reader's convenience. The edges of the cars are visible in the DVS image but barely visible in the APS image, so the RRC did not produce any detections.

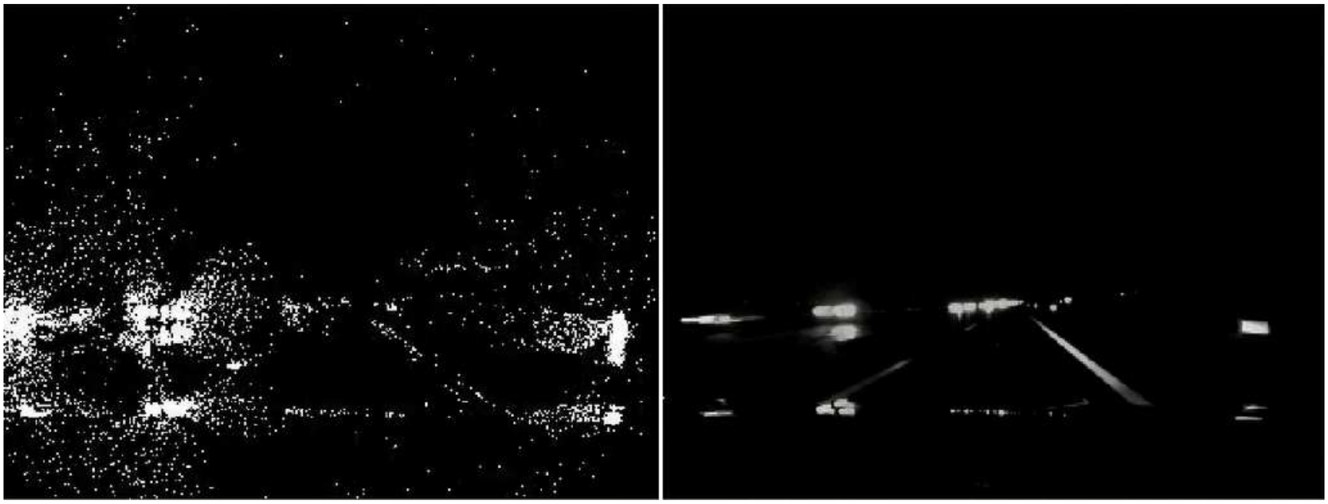

Figure 5. Left: DVS image. Right: APS image. This is a highway scene where we are only able to see the headlights of the car and nothing else. Both the DVS-only detector and the RRC fail to produce detections in such a scenario.

the order of milliWatts [5, 21], suggesting that this is another area for future research.

5. Conclusions and Future Work

In all, we have presented two main contributions. First, we showed for the first time high speed (100 FPS) object detection in a real environment under camera ego-motion, purely from dynamic vision sensor data. Previous work on event-based detection/recognition have only focused on recognizing simple objects such as numbers, or detecting objects in the absence of ego-motion, and the most realistic work is on detecting a robot in a controlled environment [15]. Our technique showed reasonable success with detections in day and night scenes, however it failed to detect cars when the headlights are bright enough to distort the features, cars which have no relative motion or cars that are too far away. Second, we showed that our trained CNN can detect cars despite motion blur and poor lighting without explicit training on such scenes, and can complement the detections from the RRC, proving that our CNN learnt robust representations of cars from pseudo-labels.

Future work includes implementing SNNs for object detection, especially on neuromorphic computing hardware. We see value in event-based image segmentation because it could boost detection performance and overcome the headlights problem in Figure 5 (*e.g.* if we detect an object on the road, then the object is more likely to be a car even though we only see headlights).

We hope that this work will encourage researchers to use pseudo-labels for supervised learning techniques on DVS data and advance the frontiers of this field, and to publish more data sets containing synchronized DVS and APS modalities.

References

[1] Y. Aytar, C. Vondrick, and A. Torralba. Soundnet: Learning sound representations from unlabeled video. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 892–900. Curran Associates, Inc., 2016.

[2] F. Barranco, C. Fermuller, Y. Aloimonos, and T. Delbruck. A dataset for visual navigation with neuromorphic methods. *Frontiers in Neuroscience*, 10:49, 2016.

[3] R. Berner, C. Brandli, M. Yang, S. C. Liu, and T. Delbruck. A 240× 180 10mw 12us latency sparse-output vision sensor for mobile applications. In *2013 Symposium on VLSI Circuits*, pages C186–C187, June 2013.

[4] J. Binas, D. Neil, S.-C. Liu, and T. Delbruck. DDD17: End-to-end DAVIS Driving Dataset. *arXiv preprint arXiv:1711.01458*, 2017.

[5] S. K. Esser, P. A. Merolla, J. V. Arthur, A. S. Cassidy, R. Appuswamy, A. Andreopoulos, D. J. Berg, J. L. McKinstry, T. Melano, D. R. Barch, C. di Nolfo, P. Datta, A. Amir, B. Taba, M. D. Flickner, and D. S. Modha. Convolutional networks for fast, energy-efficient neuromorphic computing. *Proceedings of the National Academy of Sciences*, 113(41):11441–11446, 2016.

[6] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[7] R. Ghosh, A. Mishra, G. Orchard, and N. V. Thakor. Real-time object recognition and orientation estimation using an event-based camera and CNN. In *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*, pages 544–547, Oct 2014.

[8] S. Gupta, J. Hoffman, and J. Malik. Cross modal distillation for supervision transfer. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2827–2836, June 2016.

[9] G. Hinz, G. Chen, M. Aafaque, F. Röhrbein, J. Conradt, Z. Bing, Z. Qu, W. Stechele, and A. Knoll. Online multi-object tracking-by-clustering for intelligent transportation system with neuromorphic vision sensor. In *Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz)*, pages 142–154. Springer, 2017.

[10] Y. Hu, H. Liu, M. Pfeiffer, and T. Delbruck. DVS benchmark datasets for object tracking, action recognition, and object recognition. *Frontiers in Neuroscience*, 10:405, 2016.

[11] X. Lagorce, C. Meyer, S. H. Ieng, D. Filliat, and R. Benosman. Asynchronous event-based multikernel algorithm for high-speed visual features tracking. *IEEE Transactions on Neural Networks and Learning Systems*, 26(8):1710–1720, Aug 2015.

[12] D.-H. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013.

[13] J. Li, F. Shi, W. Liu, D. Zou, Q. Wang, H. Lee, P.-K. Park, and H. E. Ryu. Adaptive temporal pooling for object detection using dynamic vision sensor. In *British Machine Vision Conf. (BMVC)*, 2017.

[14] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128×128 120 db 15μ s latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, Feb 2008.

[15] H. Liu, D. P. Moeys, G. Das, D. Neil, S. C. Liu, and T. Delbruck. Combined frame- and event-based detection and tracking. In *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2511–2514, May 2016.

[16] D. P. Moeys, F. Corradi, E. Kerr, P. Vance, G. Das, D. Neil, D. Kerr, and T. Delbruck. Steering a predator robot using a mixed frame/event-driven convolutional neural network. In *2016 Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*, pages 1–8, June 2016.

[17] 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*, 36(2):142–149, 2017.

[18] Z. Ni, S.-H. Ieng, C. Posch, S. Regnier, and R. Benosman. Visual tracking using neuromorphic asynchronous event-based cameras. *Neural Computation*, 27(4):925–953, 2015. PMID: 25710087.

[19] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer. Real-time classification and sensor fusion with a spiking deep belief network. *Frontiers in Neuroscience*, 7:178, 2013.

[20] G. Orchard, A. Jayawant, G. K. Cohen, and N. Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in Neuroscience*, 9:437, 2015.

[21] M. Osswald, S.-H. Ieng, R. Benosman, and G. Indiveri. A spiking neural network model of 3D perception for event-based neuromorphic stereo vision systems. *Scientific reports*, 7:40703, 2017.

[22] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan. Learning features by watching objects move. *arXiv preprint arXiv:1612.06370*, 2016.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[24] J. Redmon and A. Farhadi. YOLO9000: Better, faster, stronger. *arXiv preprint arXiv:1612.08242*, 2016.

[25] J. Ren, X. Chen, J. Liu, W. Sun, J. Pang, Q. Yan, Y.-W. Tai, and L. Xu. Accurate single stage detector using recurrent rolling convolution. In *CVPR*, 2017.

[26] M. Renz and Q. Wu. An energy-efficient embedded implementation for target recognition in SAR imageries. In *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–5, Nov 2017.

[27] B. Rueckauer and T. Delbruck. Evaluation of event-based algorithms for optical flow with ground-truth from inertial measurement sensor. *Frontiers in Neuroscience*, 10:176, 2016.

[28] E. Stromatias, M. Soto, T. Serrano-Gotarredona, and B. Linares-Barranco. An event-driven classifier for spiking neural networks fed with synthetic or dynamic vision sensor data. *Frontiers in Neuroscience*, 11:350, 2017.

[29] D. Tedaldi, G. Gallego, E. Mueggler, and D. Scaramuzza. Feature detection and tracking with the dynamic and active-pixel vision sensor (DAVIS). In *2016 Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*, pages 1–7, June 2016.

[30] D. Weikersdorfer, D. B. Adrian, D. Cremers, and J. Conradt. Event-based 3D SLAM with a depth-augmented dynamic vision sensor. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 359–364, May 2014.