When Do Neuromorphic Sensors Outperform cameras? Learning from Dynamic Features

Daniel Deniz
Dep. of Computer Architecture and Technology
University of Granada, Spain
danideniz@ugr.es

Eduardo Ros
Dep. of Computer Architecture and Technology
University of Granada, Spain
eros@ugr.es

Cornelia Fermüller
Perception and Robotics Group
University of Maryland, USA
fermulcm@umd.edu

Francisco Barranco
Dep. of Computer Architecture and Technology
University of Granada, Spain
fbarranco@ugr.es

2023 57th Annual Conference on Information Sciences and Systems (CISS) | 978-1-6654-5181-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/CISS56502.2023.10089678

***Abstract*—Visual event sensors only output data when changes in the scene happen at very high frequency. This allows for smartly compressing the scene and thus, enabling real-time operation. Despite these advantages, works in the literature have struggled to show a niche for these event-driven approaches compared to conventional sensors, especially when focusing on accuracy performance. In this work, we show a case that fully exploits event sensor advantages: for manipulation action recognition, learning events achieves superior accuracy and time performance. The recognition of manipulation actions requires extracting and learning features from the hand pose and trajectory and the interaction with the object. As shown in our work, approaches based on event sensors are the best fit for extracting these dynamic features contrarily to conventional approaches based on full frames, which mostly extract spatial features and need to reconstruct the dynamics from sequences of frames. Finally, we show how using a tracker to extract the features to be learned only around the hand, we obtain an approach that is scene- and almost object-agnostic and achieves good time performance with a very limited impact in accuracy.**

***Index Terms*—Event processing, manipulation action recognition**

I. Introduction

Human Action Recognition (HAR) focuses on studying human body dynamics to understand their behavior. Most commonly, HAR recognizes human activities by processing sequences of RGB frames [1]. Action recognition is a challenging problem due to the similarity-diversity duality: while different actions may look very similar (inter-class similarity), the same action may be performed very differently by different subjects (intra-class diversity). Advances in HAR are crucial for many applications such as surveillance systems [2], patient monitoring [3], or gesture recognition [4].

A special application of HAR is the identification of manipulation actions, which is of importance for the development of human-robot collaboration [5]. Due to the challenges mentioned earlier, one of the most distinctive features for action recognition is motion dynamics. Capturing motion dynamics is crucial to distinguish between manipulation actions [6], especially when subjects carry out different actions with the same object or when objects are partly occluded. In such scenarios, temporal dynamic cues become essential since using only appearance-based features can sometimes lead to incorrect classifications. However, in video-based approaches, dynamics need to be reconstructed from frames, making it difficult to capture.

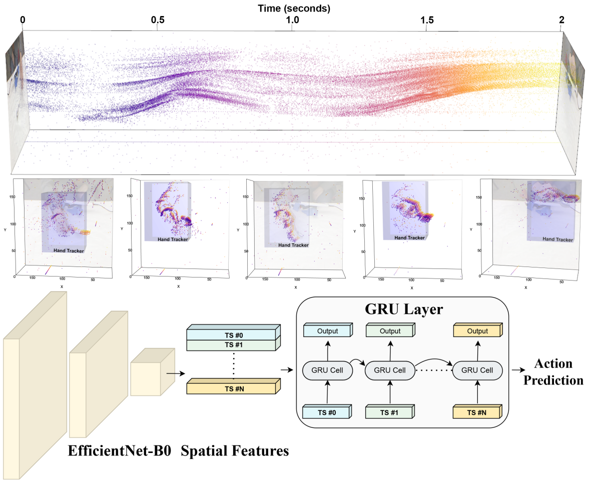


Fig. 1: An asynchronous event stream is projected into Time Surfaces while a Hand Tracker (blue bounding box) finds the hand performing the action and selects only events in its vicinity. Next, a Long-term Recurrent Convolutional Network classifies Time Surfaces to infer manipulation actions. The network comprises of two components: a) EfficientNet-B0, a 2D CNN that extracts spatial features from event time surfaces, and b) a GRU Layer that captures features of motion dynamics. The use of the hand tracker enables a significant reduction in computational complexity with a minimal impact on accuracy.

Recent solutions for HAR achieve the best results when employing Deep Learning approaches. In particular, 3D Con-

This work was supported by the Spanish National Grant PID2019-109434RA-I00/ SRA (State Research Agency /10.13039/501100011033).

978-1-6654-5181-9/23/$31.00 ©2023 IEEE

volutional Neural Networks and 2D Recurrent architectures are two of the most popular architectures for action recognition [7]. For example, in [8], the authors employ 3D CNNs for analyzing spatio-temporal features from video clips; these resource-intensive operations allow the model to simultaneously learn from the appearance and motion in the scene. In [9], authors introduce a Long-term Recurrent Convolutional Network (LRCN) that combines a 2D convolutional architecture to extract spatial features from the images and a Recurrent layer, a Long-short Term Memory (LSTM), to capture the temporal relationship of the frames in the sequence.

Neuromorphic vision sensors only output asynchronous events when changes in the scene occur, at a very high frequency. This is a smart way of compressing the scene and selecting temporal cues of moving object contours or textures [10]. Additionally, their high-temporal resolution avoids motion blur and enables real-time implementations. All these properties make event sensors the perfect candidates for activity recognition since temporal dynamics do not need to be reconstructed from sequences of frames but characterized from spatio-temporal event information. However, to do this, new event-driven approaches are needed.

In this work, we introduce an efficient event-based Deep Learning framework for action recognition. We compare it with two of the most popular Deep Learning architectures for action recognition, a 3D convolutional network, and a Long-term Recurrent Convolutional Network. The latter uses an EfficientNetB0 [11] backbone to extract spatial features combined with a Gated Recurrent Unit (GRU) network [12] to learn the temporal relationships between the frame cues. We also compare it to *Shufflenet 3D*, our implementation with 3D Convolutional layers of the *ShuffleNet v2* network [13]. Briefly, the main contributions of our work are:

- An efficient event-based Deep Learning solution for the identification of manipulation actions from asynchronous events.
- Comparison and analysis of advantages of event vision vs. RGB video approaches for action recognition.
- The use of an event-based hand tracker to reduce the computational complexity and provide an object-agnostic solution that relies only on hand dynamics

II. Related Methods

Event cameras output an asynchronous stream of events $\{\mathbf{e}_i\}_{i\in\mathbb{N}}$. Each event $\mathbf{e}_i$ is a tuple that represents the intensity change for a location at a specific time as in $\mathbf{e}_i = (x_i, y_i, t_i, p_i)$, where (x_i, y_i) are the pixel coordinates, t_i represents the timestamp at which the event was triggered, and $p_i \in \{-1, 1\}$ the event polarity (-1 or 1 for an intensity decrease or increase respectively).

Some event-driven methods use raw individual events or small packets of them to be processed in batch [14]. However, more complex processing requires more spatial and temporal support than just a few events. Then, some works propose event frames, for example, a 2D histogram that counts the events at each spatial position [15]. The main drawback of event frames is that they neglect time information. As a result, other methods in the literature also propose building Time Surfaces: 2D maps where each pixel value is a time value that provides information about when events in that spatial location occurred. For example, authors in [16] present the HOTS method, which processes packets of events in a spatio-temporal window defined by a time constant to create hierarchical Time Surfaces at different spatial scales. Moreover, they use an exponential kernel to emphasize recent events over past ones (see [17]). We build Time Surfaces with fixed spatio-temporal windows, applying an exponential decay.

A. Action Recognition

In recent years, Deep Learning has greatly advanced the state-of-the-art in Computer Vision improving performance in comparison to the use of hand-crafted features, specifically in video-based human action recognition [18]. One of the most popular Deep Learning architectures for Action Recognition from videos are 3D convolutional networks (3DCNNs). These networks use computationally-intensive 3D convolution operations on prefixed sequences of contiguous frames to extract spatio-temporal features. However, the number of frames to be analyzed in each batch might severely impact recognition if actions are carried out for longer spans of time. Another convenient solution for action identification are LRCNs, which combine 2D convolutional networks to process spatial features from the video, trained jointly with a Recurrent layer to capture temporal relationships between these spatial features.

Nowadays, several solutions for event-based action recognition have emerged. Most approaches transform asynchronous events into a spatial-grid representation in order to apply conventional computer vision algorithms. The most common approach consists in building artificial images from events and then using frame-based Convolutional Neural Networks and Recurrent Neural Networks to process sequences of events in the form of a batch of frames [19]. For example, in [20], authors propose using spatio-temporal filters to extract features from event data via convolutional layers for gesture recognition. These filters are spatio-temporal matrices learned in an unsupervised manner, showing good performance.

Despite the lack of event-driven works on manipulation action recognition, event-based human gesture recognition is a close application with many solutions using neuromorphic sensors. To accurately identify gestures, it is crucial to identify salient features to learn the simultaneous occurrence of visuospatial patterns at high speeds [21]. Some approaches have proposed architectures that do spatial feature learning with residual-graph convolutional neural networks (RG-CNN) [22] modeling temporal dependencies from asynchronous events for object classification. Other solutions use biologically plausible strategies. For example, in [21], authors use Spiking Neural Networks (SNNs) and carry out backpropagation for learning using the SLAYER frameworks. Finally, authors in [23] take advantage of the low-power consumption of event sensors, proposing an efficient system for gesture recognition on the TrueNorth neuromorphic hardware.

Finally, IBM released the *DVS 128 Gesture Recognition* dataset for human-computer interaction [23]. It includes more than 1300 samples representing 11 hand gestures performed by 29 subjects. However, this dataset does not include any human-object interaction. That is the reason for us to also train our approach on the *Event Manipulation Action Dataset* (*E-MAD*) [24]. It includes 750 samples of 5 subjects carrying out 30 different actions interacting with 6 objects.

III. Our Approach

In this work, we introduce an efficient Deep Learning framework for event-based action recognition. We analyze the performance of two of the most popular architectures for action recognition, a 3D convolutional network, and a Long-term Recurrent Convolutional Network. Then, we assess our event-driven solution on two datasets for action recognition. Also, we compare our event-driven solution with the RGB video-based approach using the *Manipulation Action Dataset* (MAD) [25], analyzing the advantages of neuromorphic vision sensors for action recognition.

In a second stage, we integrate a hand-tracking method for event-based data [26]. The integration of the hand tracker in the pipeline for action recognition seeks two objectives: 1) reducing the number of events to be analyzed by the neural network, limiting the region of interest to the area containing the hand that manipulates the object; 2) analyzing the ablation study with an (almost) object-agnostic model. The first objective is relevant for the reduction of the computational complexity and thus, exploiting the high-temporal resolution of event processing with the network that integrates the knowledge from every event as soon as it is processed. The second objective also helps to understand if the information from the events that characterize the hand motion, i.e. the hand dynamics, is enough for the prediction and what is the impact of using only these features to perform recognition.

A. Manipulation Action Recognition

Firstly, to process asynchronous event information, we build grid-like representation structures. In particular, we follow the exponentially decaying factor procedure to build Time Surfaces as done in [17]. This method maps a group of events within a time slice (of a predefined length) into a 2D spatial representation, with a decaying factor that uses larger weights for more recent events.

Action Recognition via Deep Learning: In this work, we present two alternatives for event-based action recognition. One of the solutions is a 3DCNN architecture. This network processes a batch of Time Surfaces simultaneously to extract spatio-temporal features from neuromorphic data. 3DCNN are very complex architectures that are hard to optimize due to their large number of parameters. The implementation uses a *ShuffleNet v2* network [13] as the backbone of our 3DCNN architecture (*Shufflenet 3D*). The *Shufflenet* is a more efficient approach compared to other state-of-the-art solutions such as ResNet [27] or Inception [28] networks. It reduces element-wise operations and performs a channel shuffle of the convolutional operations aiming to enable communication between groups of channels to increase accuracy and boost efficiency [13].

On the other hand, we also compare with an LRCN that combines a 2D CNN and an RNN. The 2D CNN extracts a feature vector from the input Time Surfaces. Then, the RNN analyzes the sequence of feature vectors extracted from contiguous Time Surfaces, in order to learn from the motion dynamics to identify manipulation actions. This architecture is trained end-to-end, meaning that the 2DCNN and the RNN module are trained simultaneously. In particular, we selected an *EfficientNet-B0* architecture [11] for our 2D CNN part. The *EfficientNet* is a family of models designed following a new scaling method that uniformly scales all dimensions depth/width/resolution, defined as compound scaling. This architecture offers state-of-the-art performance while being 8.4x smaller and 6.1x faster than other state-of-the-art CNNs [11]. The *EfficientNet-B0* is the most simple and efficient model of the EfficientNet family. Regarding the RNN part, we integrate a GRU layer [12]. GRUs show better performance in large sequences compared to conventional RNNs.

Event Manipulation Action Dataset: We train our solutions on two event-based datasets: the *DVS128 gesture dataset* [23] and the *Event Manipulation Action Dataset* or *E-MAD* [24]. The *DVS128 gesture dataset* includes 11 half-body gestures of 29 subjects mostly done with their hands and arms. They were recorded under 3 different illumination conditions. Next, the *E-MAD* includes 750 samples representing 30 manipulation actions using 6 different objects and performed by 5 subjects. The objects considered are: a *Cup*, a *Stone*, a *Sponge*, a *Spoon*, a *Knife*, and a *Spatula*.

Training procedure: Given the short actions in the datasets, our approach builds new Time Surfaces every 33 ms. Then, training is done as follows: first, different data augmentation techniques such as random rotation or random time window are applied, and then Time Surfaces are built. Afterward, we only train the final layers of the network for 60 epochs (we use pre-trained weights from ImageNet), and finally, the model is trained end-to-end for another 60 epochs. For the *E-MAD*, training is done in three phases: 1) the model is trained for 100 epochs to distinguish only between the 6 super-categories of actions on objects, to guide the optimization of the architecture due to the low number of samples per action; 2) transfer learning is applied for 10 epochs to identify the 30 manipulation actions; 3) the model is fine-tuned by training it end-to-end for 100 additional epochs. During this procedure, we use Adam optimizer with a starting learning rate of 0.0003. The learning rate is decayed by a 0.1x factor when validation loss reaches a plateau for at least 5 epochs.

B. Hand tracker

As part of our study, an event-based hand tracker [26] is integrated into the pipeline. This tracker is a very fast method that implements a cluster tracker that assumes spatially-connected events in rectangular areas as events that belong to the same object part. This simple method only requires a

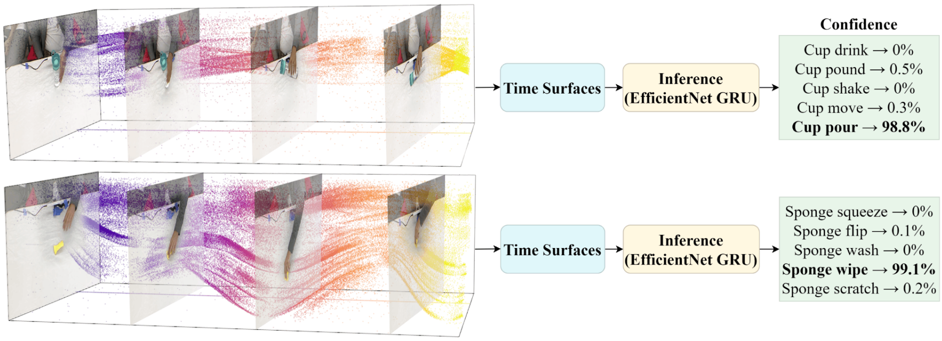


Fig. 2: Example of classifications of two samples from *E-MAD*. Asynchronous events are projected into Time Surfaces and then, the confidence predicted by the *EfficientNet GRU* is listed. Top: *pouring a cup*; bottom: *wiping with a sponge*.

TABLE I: Accuracy of models on DVS 128 Gestures

Architecture	Accuracy
RG-CNN [22]	97.20%
TORE [29]	96.20%
EvT [30]	96.20%
Shufflenet v2 3D	94.31%
EfficientNet GRU	97.34%

first input that is used to build the model of the object to be tracked. Moreover, it runs in real-time since only pixels that generate events need to be processed within the region of interest. The computational cost is limited to the search of the nearest cluster that is generally very close due to the high-temporal resolution of the neuromorphic sensor.

The hand tracker locates and tracks the hand that manipulates the object and performs the action, limiting the data to be processed to the area around the hand. In this case, our hypothesis is that the action should be identified using only the hand dynamics, and therefore, events from this region will suffice for this task. Note though that, after the contact point (when the hand touches the object), events that belong to the object itself are partially contained in the hand region since the hand and object cannot be split apart from there on.

The aim of using the hand tracker to select the region containing the hand performing the action is two-folded. On the one hand, reducing the spatial resolution of the input time surface has a direct impact on the time performance of the prediction. The resolution reduction limits the floating-point operations (FLOPS) required to perform inference, thus making the prediction faster. On the other hand, restricting the input to the hand data helps the model to generalize more effectively since other details about the subject or the scene are left out. Furthermore, using only the output from the hand tracker favors the model to focus on the hand dynamics.

IV. Discussion and Results

In this section, we first present an evaluation of our event-based solutions on the *DVS128 Gesture Dataset* [23] and the *E-MAD* dataset. Next, we analyze the benefits of event-based approaches versus video-based approaches. Finally, we discuss the use of the hand tracker, comparing the different alternatives and assessing the impact in terms of computational complexity/time performance and accuracy.

TABLE II: 4-Fold Cross Validation Manipulation Action classification

DL Architecture	Video-based Accuracy	Event-based Accuracy
EfficientNet GRU	79.02 ± 7.07	**97.63 ± 0.94**

A. Event-based Action Recognition

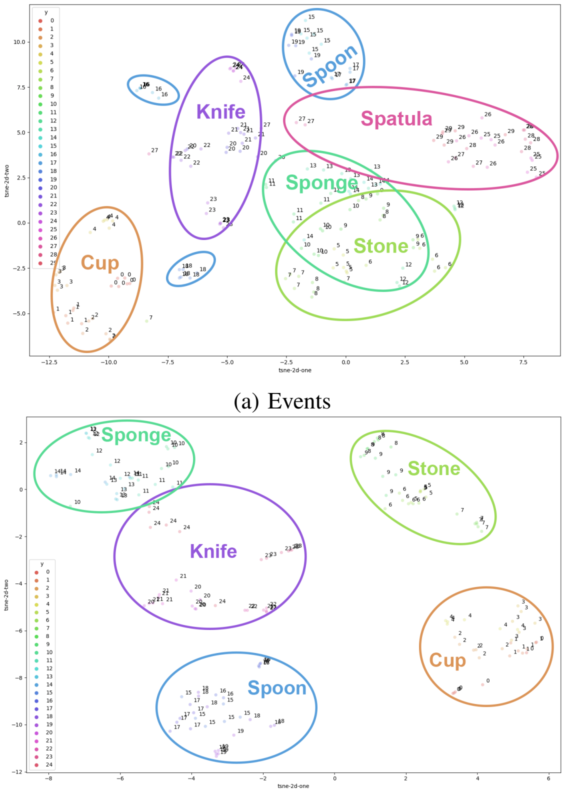
Table I shows the performance of different DL solutions trained for gesture recognition and compares it with our two proposals. The state-of-the-art works in the top part reach an accuracy above 96%, particularly RG-CNN [22] achieves 97.2%. This is an approach based on residual-graph convolutional neural networks that takes advantage of the sparse and asynchronous nature of event data. Next, on the bottom part of the Table, we show the accuracy for our *Shufflenet v2 3D* and *EfficientNet GRU* implementations. Note, for example, how the *Shufflenet v2 3D* offers a great performance above 94%, but still about 2 points less than the other alternatives. Conversely, the *EfficientNet GRU* reaches the highest accuracy for gesture recognition, making it the best candidate for action recognition.

However, results in Table I show one of the main problems with the *DVS128 Gesture Dataset*. Most methods reach over 95% accuracy because of the inter-action great variation which makes it easy to recognize the action by just focusing on the spatial features of event activity in certain regions of the sensor image plane. In fact, the action dynamics do not seem to play a relevant role in the different sequences; the main challenge seems to be the intra-action illumination condition differences.

In order to expose the potential of event data, the recognition problem must require the learning of the action dynamics, not only focusing on spatial features. Next, we assess the importance of event-based vision for the recognition of manipulation actions. When identifying manipulation actions, the subject interacts with objects performing different actions. Given the inter-class similarities when carrying out different actions with the same object, accurately capturing the action dynamics is essential for its recognition.

TABLE III: Hand Tracker performance comparison: 4-Fold Cross Validation E-MAD

DL architecture	# Params (Millions)	Full resolution			Using hand tracker		
		GFlops	TS/s	Accuracy	GFlops	TS/s	Accuracy
EfficientNet GRU	4.77	10.34	3823	97.63 ± 0.94	6.42	6062	93.96 ± 2.99



(a) Events

(b) Videos

Fig. 3: t-SNE projection of features extracted by the *EfficientNet GRU*. a) Using events that provide accurate cues of the scene dynamics. However, the lack of appearance-based features causes samples not only to be categorized into classes in terms of which object is taking part in the action but also the particular motion. b) Using videos, appearance features significantly contribute to distinguishing between objects. However, classes of actions on objects are not clearly separated.

Regarding action recognition using videos, previous works have shown how the spatial features from the object used in the action are decisive for recognition [6]. To compare the performance of event-based vision with video-based approaches, we train our *EfficientNet GRU* model on *E-MAD* and the *Manipulation Action Dataset* (*MAD*) [25]. *MAD* and *E-MAD* were recorded simultaneously with a conventional camera and a neuromorphic sensor, respectively. Taking this into account, they both show the same subjects performing the actions. Some examples are shown in Figure 2, which also lists the confidence reached for the top-5 labels.

Figure 3 along with Table II support the claim that video-based approaches focus on spatial features and on reconstructing the action dynamics from frames that do not suffice for action manipulation recognition. Contrarily, event-driven solutions do extract features from the action dynamics without requiring its reconstruction from synchronous samples, favoring better recognition. The Figure presents two t-Distributed Stochastic Neighbour Embeddings (t-SNE) [31], projecting the features extracted by the DL models onto a 2D space. Figure 3b shows the 2D projection for the video-based solution. Note how the actions that involve the same object are clearly clustered. This is because appearance provides enough information to identify which object is the subject using. In the case of the event-based solution (see Figure 3a), objects are not as clearly separated because motion provides more information than appearance when identifying actions. Table II shows how the event-based alternative is around 23% more accurate than the video-based solution. Despite the clear clustering shown in the t-SNE figures for the objects, this table shows that video-based approaches fail at differentiating more subtle cues in the motion dynamics, leading to a worse accuracy. However, with event-driven solutions, the recognition lies in action dynamics, capturing them more accurately without the need of reconstructing them from synchronous samples.

B. *Time Performance: Hand Tracker*

In this last experiment, a hand-tracker was integrated into the pipeline to understand if the information used for classification is limited to the hand motion and its interaction with the object. In other words, since these are manipulation actions, if the cues from the action dynamics are enough for recognition. Moreover, limiting the area to be processed benefits the final recognition, improving inference time performance, and more importantly for human-robot interaction applications, it also reduces the overall system latency.

In our analysis, we perform an ablation comparing the *EfficientNet GRU* models using the full frame (left column) and reducing the input to the hand region (right column) integrating the hand tracker into the pipeline. Table III summarizes the results following a K-fold ($K = 4$) cross validation. It includes the overall system performance in Giga floating-point operations (Giga Flops), the inference time performance in Time Surfaces processed per second, and the accuracy.

The left column shows that the *EfficientNet GRU* model reaches an average accuracy above 97% when using full resolution, requiring more than 10 GFlops to perform inference. However, limiting the input events to the area around the hand (right column) results in 38% computational reduction, leading also to x1.6 faster inferences (TS/s). Furthermore, this is achieved with a limited impact on the accuracy of only 3.67 points. Therefore, this experiment shows that recognition mostly lies in the cues from the hand performing the action despite the challenging object-hand occlusions.

V. Conclusions

For years, conventional frame-based methods have always outperformed neuromorphic approaches, particularly with regard to accuracy performance. The work presented in this paper shows that action recognition is one of the problems for which event-driven solutions achieve better results. Moreover, event-driven solutions offer other benefits such as real-time operation and low latency due to their intrinsic properties. These advantages are highly valuable in fields such as human-robot interaction which precisely requires action recognition.

This work shows how event-driven neural networks are able to learn spatio-temporal features to achieve successful recognition, in contrast to video-based approaches. The experimental work conducted in this paper shows that the reconstruction required in video-based approaches is not enough to capture action dynamics. Additionally, it proves that video-based approaches focus on spatial features, which do not suffice for successful recognition. This is clearer when analyzing challenging manipulation actions with the same object, where differences are found in the action itself.

Finally, the integration of the hand tracker also shows that our event-driven approach is scene- and almost object-agnostic. Moreover, learning only features around the manipulating hand considerably improves time performance while the impact on accuracy is very limited.

References

[1] N. Gupta, S. K. Gupta, R. K. Pathak, V. Jain, P. Rashidi, and J. S. Suri, "Human activity recognition in artificial intelligence framework: a narrative review," *Artificial Intelligence Review*, pp. 1–54, 2022.

[2] R. Khurana and A. K. S. Kushwaha, "Deep learning approaches for human activity recognition in video surveillance-a survey," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*. IEEE, 2018, pp. 542–544.

[3] D. Deniz, F. Barranco, J. Isern, and E. Ros, "Reconfigurable cyber-physical system for lifestyle video-monitoring via deep learning," in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1. IEEE, 2020, pp. 1705–1712.

[4] O. K. Oyedotun and A. Khashman, "Deep learning in vision-based static hand gesture recognition," *Neural Computing and Applications*, vol. 28, no. 12, pp. 3941–3951, 2017.

[5] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Autonomous Robots*, vol. 42, no. 5, pp. 957–975, 2018.

[6] C. Fermüller, F. Wang, Y. Yang, K. Zampogiannis, Y. Zhang, F. Barranco, and M. Pfeiffer, "Prediction of manipulation actions," *International Journal of Computer Vision*, vol. 126, no. 2, pp. 358–374, 2018.

[7] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," *International Journal of Computer Vision*, vol. 130, no. 5, pp. 1366–1401, 2022.

[8] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[9] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[10] F. Barranco, C. Fermüller, and Y. Aloimonos, "Contour motion estimation for asynchronous event-driven cameras," *Proceedings of the IEEE*, vol. 102, no. 10, pp. 1537–1556, 2014.

[11] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[12] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[13] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[14] E. Mueggler, G. Gallego, H. Rebecq, and D. Scaramuzza, "Continuous-time visual-inertial odometry for event cameras," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1425–1440, 2018.

[15] G. Gallego, T. Delbrück, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis *et al.*, "Event-based vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 1, pp. 154–180, 2020.

[16] 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.

[17] S. Afshar, N. Ralph, Y. Xu, J. Tapson, A. v. Schaik, and G. Cohen, "Event-based feature extraction using adaptive selection thresholds," *Sensors*, vol. 20, no. 6, p. 1600, 2020.

[18] G. Yao, T. Lei, and J. Zhong, "A review of convolutional-neural-network-based action recognition," *Pattern Recognition Letters*, vol. 118, pp. 14–22, 2019.

[19] F. J. Moreno-Rodríguez, V. J. Traver, F. Barranco, M. Dimiccoli, and F. Pla, "Visual event-based egocentric human action recognition," in *Iberian Conference on Pattern Recognition and Image Analysis*. Springer, 2022, pp. 402–414.

[20] R. Ghosh, A. Gupta, A. Nakagawa, A. Soares, and N. Thakor, "Spatiotemporal filtering for event-based action recognition," *arXiv preprint arXiv:1903.07067*, 2019.

[21] A. Vasudevan, P. Negri, B. Linares-Barranco, and T. Serrano-Gotarredona, "Introduction and analysis of an event-based sign language dataset," in *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*. IEEE, 2020, pp. 675–682.

[22] Y. Bi, A. Chadha, A. Abbas, E. Bourtsoulatze, and Y. Andreopoulos, "Graph-based spatio-temporal feature learning for neuromorphic vision sensing," *IEEE Transactions on Image Processing*, vol. 29, pp. 9084–9098, 2020.

[23] 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.

[24] D. Deniz, E. Ros, C. Fermuller, and F. Barranco, "Event-based vision for early prediction of manipulation actions," unpublished.

[25] C. Fermüller, F. Wang, Y. Yang, K. Zampogiannis, Y. Zhang, F. Barranco, and M. Pfeiffer. (2018) Prediction of manipulation actions. [Online]. Available: http://users.umiacs.umd.edu/~fer/action-prediction/

[26] T. Delbruck, "Frame-free dynamic digital vision," in *Proceedings of Intl. Symp. on Secure-Life Electronics, Advanced Electronics for Quality Life and Society*, vol. 1. Citeseer, 2008, pp. 21–26.

[27] 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.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[29] R. Baldwin, R. Liu, M. Almatrafi, V. Asari, and K. Hirakawa, "Time-ordered recent event (tore) volumes for event cameras," *arXiv preprint arXiv:2103.06108*, 2021.

[30] A. Sabater, L. Montesano, and A. C. Murillo, "Event transformer. a sparse-aware solution for efficient event data processing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2677–2686.

[31] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.