Denoising of event-based sensors with deep neural networks

Zhihong Zhang[a], Jinli Suo[a, †], and Qionghai Dai[a]

[a]Department of Automation, Tsinghua University, Beijing 100084, China

ABSTRACT

As a novel asynchronous imaging sensor, event camera features low power consumption, low temporal latency and high dynamic range, but abundant noise. In real applications, it is essential to suppress the noise in the output event sequences before successive analysis. However, the event camera is of address-event-representation (AER), and requires developing new denoising techniques rather than conventional frame-based image denoising methods. In this paper, we propose two learning-based methods for the denoising of event-based sensor measurements, i.e., convolutional denoising auto-encoder (ConvDAE) and sequence-fragment recurrent neural network (SeqRNN). The former converts the event sequence into 2D images before denoising, which is compatible with existing deep denoisers and high-level vision tasks. The latter, utilizes recurrent neural network's advantages in dealing with time series to realize online denoising while keeping the event's original AER representation. Experiments based on real data demonstrate the effectiveness and flexibility of the proposed methods.

Keywords: Event-based sensors, denoising, recurrent neural network, convolutional denoising auto-encoder

1. INTRODUCTION

With the development of neuromorphic vision, event-based sensors, also known as "silicon retina", have become popular in recent years. Inspired by the working mechanism of human retina, the sensor is only sensitive to the region with brightness changes and outputs event sequences containing information of changing positions, time and polarities in a special format named address-event-representation (AER). Compared with traditional cameras, event cameras have many advantages, such as low power consumption (23 mW), low temporal latency(15 μs) and high dynamic range (>120 dB),[1] which provide them with great potential in auto-driving,[2–4] high-speed target detection and tracking,[5–8] simultaneous localization and mapping (SLAM)[9–11] and so on. However, the low signal to noise ratio (SNR) of original event sequences seriously limits event cameras' applications in actual scenarios. Besides, due to event cameras' unique data presentation, i.e., AER, it is hard to directly adapt traditional frame-based denoising methods to the event camera.

Most existing denoising methods for event cameras leverage the spatial-temporal correlation of the events to filter out the unpleasant noise. For example, in 2015, Liu *et al*. designed a 1 mW, 10 ns-latency mixed signal system in 0.18 μm CMOS to filter out uncorrelated background activity in event-based sensors, realizing a removal rate of 98% for the background noise in real time.[12] In 2016, Czech *et al*. quantitatively evaluated 8 different filtering algorithms and emphasized the importance of the activity of neighbouring pixels consistency in filtering performance.[13] Other methods relying on dictionary learning[14] or neural networks on neuromorphic chips[15] were also proposed to denoise the noisy event sequences. However, most of the existing methods are based on heuristically designed filters or shallow learned features, which can hardly cope with complicated noise adaptively in different scenes.

In recent years, deep learning based approaches have been widely used in many vision tasks and achieved great success. As a data-driven method, deep neural networks automatically learn the deep features from massive data to perform sophisticated inference and have better generalization ability compared with traditional methods. To bridge the gap between current event-based sensors and real scenario applications, we proposed two deep neural network based denoising methods for event-based sensors, i.e., convolutional denoising auto-encoder and sequence-fragment recurrent neural network, which are suitable for different vision tasks. Real data experiments demonstrate that proposed methods can successfully remove the noise in the event camera data, and meet the application requirements in corresponding scenarios.

† jlsuo@tsinghua.edu.cn

2. METHODS

Recently, many works combining event cameras and deep learning have emerged to cope with high-level vision tasks and realized great success. Among these works, there are mainly two streams based on convolutional neural networks (CNNs), and recurrent neural networks (RNNs), respectively. In this paper, we hope to propose learning-based light-weight deep denoisers which can be easily integrated into aforementioned high-level tasks' solutions. Thus, a CNN-based convolutional denoising auto-encoder and a RNN-based sequence-fragment recurrent neural network are designed below to adapt to image-based and event-based downstream tasks' solutions respectively.

2.1 Convolutional Denoising Auto-Encoder

In recent decades, convolutional neural network have been popularized to perform various computer vision tasks, such as image restoration,[16–18] classification,[19–21] object detection[22–24] and so on, and bring about a great boost for the computer vision community. Though CNNs show prominent superiority in extracting image or video features and conducting corresponding inference, the output of event cameras is a stream of events in AER format, applying conventional denoising techniques to process event camera measurements is nontrivial. In order to keep compatible with CNN-based denoising approaches, an intuitive way is to map the event sequence into a series of video frames according to the position and time stamp contained in each event point. Specifically, a temporal sliding window with a presetting step is applied to the event sequence. In each step, a video frame will be generated by mapping the events containing in the current window to corresponding pixel positions, and the intensity of the pixel equals to the polarity of the latest event appearing at that position. By changing the length and step of the sliding window, we can control the frame rate and temporal smoothness/consistency of the generated video.

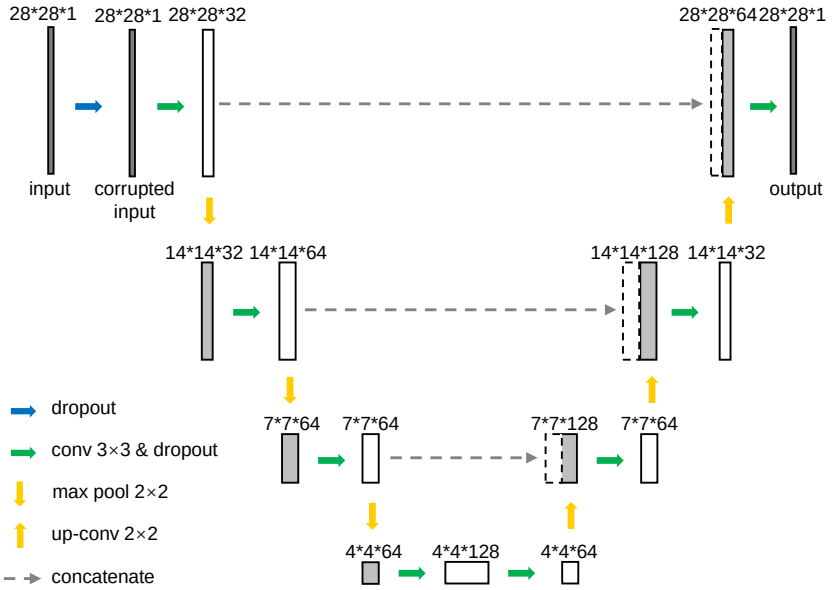

Figure 1. The architecture of the proposed self-supervised convolutional denoising auto-encoder (ConvDAE).

After converting the event sequence into a video, we can make use of existing CNNs for the successive denoising process. It is worth noting that the denoising for event cameras is an unsupervised task, since clean event sequences are inaccessible for existing event cameras. Recently, self-supervised denoising methods like Noise2Self[17] and Noise2Void[25] have been proposed and demonstrate excellent performance in real applications. In these works, a "corrupted" or "noisy" image and the "original" image can serve as a pair of training data, guiding the network to learn the latent image prior for denoising. Bearing this in mind, we build a self-supervised convolutional denoising auto-encoder, dubbed ConvDAE as illustrated in Fig. 1. The basic structure of ConvDAE

is a simplified U-Net, with only one convolutional layer in each encoding or decoding layer, and there are four pairs of encoding and decoding layers in total. In ConvDAE, we use a dropout layer to randomly corrupt the original image before inputting it into the simplified U-Net. Meanwhile, we further conduct a pre-denoisng process on the source events corresponding to the original image, and then generate a relatively clean image from the pre-denoised events as the "ground truth", instead of directly using the original image to calculate the loss as other works do. The pre-denoising is based on a nearest neighbor filtering (NNF) algorithm,[13] and can provide the network with a stronger guidance for the denoising. Finally, back propagation can be performed with the mean square error (MSE) between the network's output and the "ground truth" image as the loss.

2.2 Sequence-Fragment Recurrent Neural Network

Mapping the event sequence into a conventional video provides a direct way to leverage CNNs for event camera's denoising, and can make event cameras compatible with existing high-level vision tasks' solutions. However, the mapping operation is irreversible, as it discards the accurate time stamp for the events and gather them together into one frame. Although we can flexibly adjust the length and step of the sliding window to control the property of the generated video during mapping, this approach will inevitably sacrifice the temporal resolution of event cameras to some extent and change their original data format of AER. So in this subsection, we regard the denoising of event sequence as a temporal series processing problem, and leverage a recurrent neural network[26] based architecture to finish the task. In this way, we can keep the original data format of the events, and neural networks suitable for temporal series processing like the spike neural network (SNN)[27] and RNN can be utilized to perform successive high-level tasks.

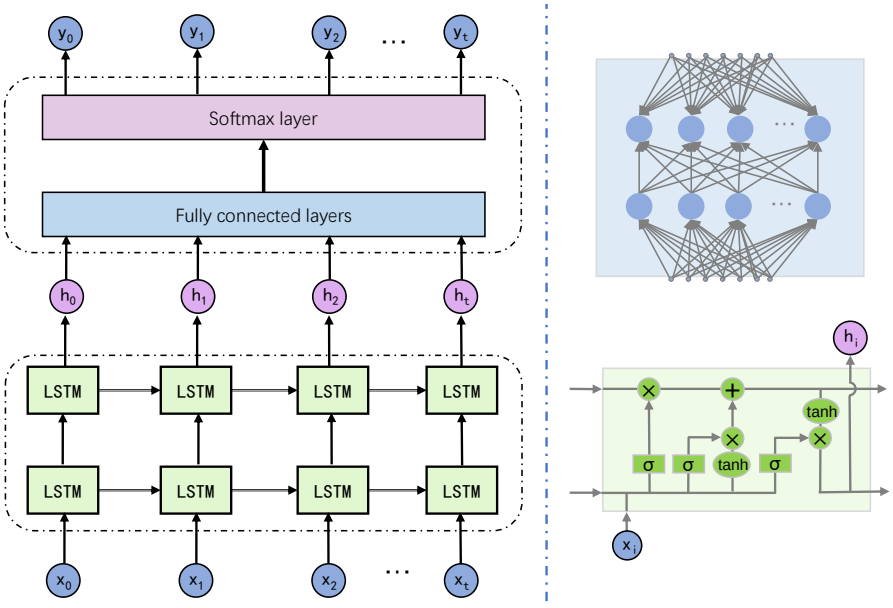

Figure 2. The architecture of the proposed sequence-fragment recurrent neural network (SeqRNN).

There are currently many RNN-based network architectures like basic recurrent neural networks (RNN)[26] , long short-term memory (LSTM)[28] , bi-directional long short-term memory (BiLSTM)[29] and gated recurrent units (GRU).[30] After extensive experiments, we choose the widely used LSTM as our basic network structure, and propose sequence-fragment recurrent neural network (SeqRNN) for the denoising of event sequence. Here, we model the denoising of events as a classification problem. For each event in the input sequence, SeqRNN will output a corresponding classification tag as -1, 0 or 1, which is corresponding to a negative event, noise or a positive event respectively. As is shown in Fig. 2, the overall structure of SeqRNN can be divided into two parts. The first part consists of two LSTM layers, which are used to dig out the features across a segment of consecutive event sequence. And the second part is composed of three fully connected layers and a softmax layer,

which integrates all the hidden statuses from the second LSTM layer and finish the classification of the input events. During training, we divide the event sequence into short segments with the same length, and input them into SeqRNN sequentially. Similar to what we did in ConvDAE, a pre-denoising module based on the nearest neighbor filtering algorithm is introduced to generate a relatively clean event sequence in accord with the input sequence, to serve as the "ground truth" and provide guiding prior for the unsupervised training process. We use the cross-entropy between the "ground truth" and SeqRNN's outputs as the network's loss during training.

3. RESULTS

In this section, we describe the dataset preparation and implementation detailes in Sec. 3.1, and report our experimental results of ConvDAE and SeqRNN in Sec. 3.2 and Sec. 3.3, respectively.

3.1 Dataset and Implementation

Dataset plays an important role in the boom of deep learning. However, creating dataset for event-based sensors remains a challenge, considering that event cameras are mainly used in industrial applications and not so popularized as conventional cameras in our daily life. In this paper, we demonstrate the effectiveness of proposed ConvDAE and SeqRNN with N-MNIST dataset,[31] which is converted from classical static computer vision image dataset MNIST.[32] As mentioned in Sec. 1, event cameras are only sensitive to brightness change, and have little response to static scenes. So N-MNIST is collected by capturing the MNIST images displayed on a screen with a event camera mounted on a moving stage, which moves periodically with assigned trajectory and speed during capturing. There are 70k event sequences in N-MNIST dataset, which are divided into 60k and 10k for training and testing respectively.

The neural networks are implemented with TensorFlow, and the training is conducted on a work station equipped with a NVIDIA 1080Ti GPU. We choose Adam as our optimizer, and set the batch size to 128. The initial learning rate is set to 10^{-4}, and learning rate decay strategy is employed during the training.

3.2 Results for ConvDAE

As mentioned in Sec. 2.1, we firstly convert the event sequences contained in N-MNIST dataset into two dimensional frames before dealing with ConvDAE. For each event sequence, we use a sliding window with 40 ms length and 10 ms step to generate successive frames. Besides, the frames containing few contents are discarded. The final evaluation results of ConvDAE are summarized in Fig. 3. The first row shows ten typical frames converted from original event sequences, the second row illustrates the denoising results of the first row by ConvDAE, and the third row serves as the reference images which are generated from the pre-denoised events. As can be see from Fig. 3, ConvDAE can efficiently remove the dispersed noise shown in the original images, while keeping the useful signals. And compared with the reference images which are based on the nearest neighbor filtering algorithm, ConvDAE performs better in distinguishing the signal and noise especially around the objects.

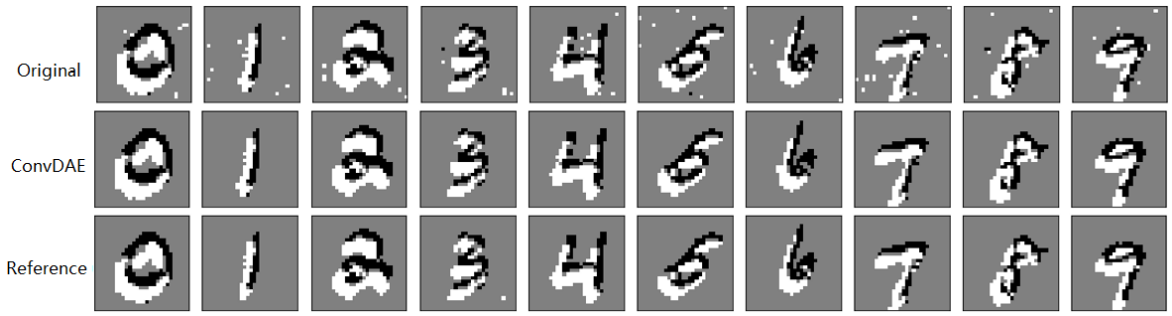

Figure 3. The denoising results of convolutional denoising auto-encoder (ConvDAE). **Original**: the images mapped from original event sequences; **ConvDAE**: the denoising results of the original images based on ConvDAE; **Reference**: the reference images mapped from the pre-denoised event sequences via nearest neighbor filtering.

3.3 Results for SeqRNN

In the evaluation of SeqRNN, we divide the event sequences into short segments, each containing 64 event points. The segment length is determined by the specific scene, and the criterion is to ensures that each segment contains enough semantic information. In this way, each segment can provide adequate scene prior for the training of SeqRNN. In order to visualize the denoising events, we mapping the events in each segment into a single image. Ten typical event segments and corresponding denoising results are visualized in the first row and the second row of Fig. 4. The images generated from the nearest neighbor filtering pre-denoised events are illustrated in the third throw of Fig. 4 for reference. By comparison, we can find that, the noise shown in the original event segments can be filtered out clearly by SeqRNN, which meanwhile outperforms the nearest neighbor filtering algorithm in some details.

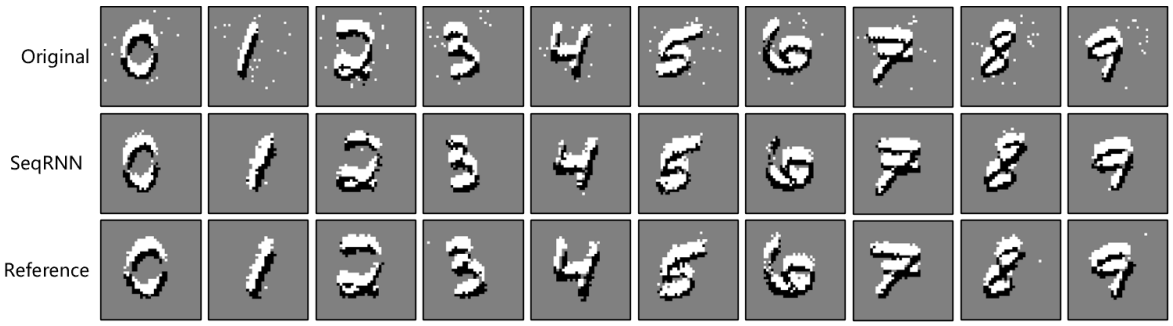

Figure 4. The denoising results of sequence-fragment recurrent neural network (SeqRNN). **Original**: the visualization of original event sequences; **SeqRNN**: the visualization of the denoising results of original event sequences based on SeqRNN; **Reference**: the reference images visualized from the nearest neighbor filtering pre-denoised event sequences

4. CONCLUSION

In this paper, we come up with two different learning-based methods named ConvDAE and SeqRNN for the denoisng of event camera data, and demonstrate their effectiveness and flexibility with real data experiments. ConvDAE converts the event sequences into video frames, and then leverages a self-supervised convolutional denoising auto-encoder for the denoising of the noisy images. SeqRNN takes advantage of RNNs' abilities in dealing with temporal series to remove the noisy events in the event sequence while keep it is original data format of AER. In a nutshell, ConvDAE and SeqRNN use two different network architectures of CNN and RNN realizing the denoising of event cameras, which makes them compatible with different high-level vision tasks' solutions. In the future, we will test the performance of the proposed ConvDAE and SeqRNN on more complicated datasets, and also evaluate their denoising performance based on their assistance for the downstream high-level vision tasks, such as classification, detection, tracking and *etc*.

ACKNOWLEDGMENTS

This work is jointly funded by Ministry of Science and Technology of China (Grant No. 2020AA0108200), National Natural Science Foundation of China (Grant No. 61931012 and 62088102) and Beijing Natural Science Foundation (Grant No. Z200021).

REFERENCES

[1] Lichtsteiner, P., Posch, C., and Delbruck, T., "A 128×128 120 dB 15us Latency Asynchronous Temporal Contrast Vision Sensor," *IEEE Journal of Solid-State Circuits* **43**(2), 566–576 (2008).

[2] Hu, Y., Binas, J., Neil, D., Liu, S.-C., and Delbruck, T., "DDD20 End-to-End Event Camera Driving Dataset: Fusing Frames and Events with Deep Learning for Improved Steering Prediction," in [*International Conference on Intelligent Transportation Systems (ITSC)*], 1–6, IEEE, Rhodes, Greece (2020).

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

[4] Moeys, D. P., Corradi, F., Kerr, E., Vance, P., Das, G., Neil, D., Kerr, D., and Delbruck, T., "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)*], 1–8, IEEE, Krakow, Poland (2016).

[5] Renner, A., Evanusa, M., and Sandamirskaya, Y., "Event-based attention and tracking on neuromorphic hardware," in [*2019 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*], 1709–1716, IEEE (2019).

[6] Huang, J., Wang, S., Guo, M., and Chen, S., "Event-Guided Structured Output Tracking of Fast-Moving Objects Using a CeleX Sensor," *IEEE Transactions on Circuits and Systems for Video Technology* **28**(9), 2413–2417 (2018).

[7] Jiang, R., Mou, X., Shi, S., Zhou, Y., Wang, Q., Dong, M., and Chen, S., "Object tracking on event cameras with offline–online learning," *CAAI Transactions on Intelligence Technology* **5**(3), 165–171 (2020).

[8] Wang, Y., Idoughi, R., and Heidrich, W., "Stereo event-based particle tracking velocimetry for 3D fluid flow reconstruction," in [*Euroapean Conference on Computer Vision (ECCV)*], 36–53, Springer, Springer International Publishing, Cham (2020).

[9] Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., and Scaramuzza, D., "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).

[10] Rebecq, H., Horstschäfer, T., Gallego, G., and Scaramuzza, D., "Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time," *IEEE Robotics and Automation Letters* **2**(2), 593–600 (2016).

[11] Vidal, A. R., Rebecq, H., Horstschaefer, T., and Scaramuzza, D., "Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High-Speed Scenarios," *IEEE Robotics and Automation Letters* **3**(2), 994–1001 (2018).

[12] Liu, H., Brandli, C., Li, C., Liu, S.-C., and Delbruck, T., "Design of a spatiotemporal correlation filter for event-based sensors," in [*International Symposium on Circuits and Systems (ISCAS)*], 722–725, IEEE, Lisbon, Portugal (2015).

[13] Czech, D. and Orchard, G., "Evaluating noise filtering for event-based asynchronous change detection image sensors," in [*IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*], 19–24, IEEE, Singapore, Singapore (2016).

[14] Xie, X., Du, J., Shi, G., Hu, H., and Li, W., "An Improved Approach for Visualizing Dynamic Vision Sensor and its Video Denoising," in [*International Conference on Video and Image Processing - ICVIP 2017*], 176–180, ACM Press, Singapore, Singapore (2017).

[15] Padala, V., Basu, A., and Orchard, G., "A Noise Filtering Algorithm for Event-Based Asynchronous Change Detection Image Sensors on TrueNorth and Its Implementation on TrueNorth," *Frontiers in Neuroscience* **12**, 118 (2018).

[16] Zhang, K., Zuo, W., Gu, S., and Zhang, L., "Learning Deep CNN Denoiser Prior for Image Restoration," in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 2808–2817, IEEE, Honolulu, HI (2017).

[17] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T., "Noise2Noise: Learning image restoration without clean data," in [*International Conference on Machine Learning*], 2971–2980 (2018).

[18] Deng, X. and Dragotti, P. L., "Deep Convolutional Neural Network for Multi-Modal Image Restoration and Fusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **43**(10), 3333–3348 (2021).

[19] Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D. D., and Chen, M., "Medical image classification with convolutional neural network," in [*International Conference on Control Automation Robotics & Vision (ICARCV)*], 844–848, IEEE, Singapore (2014).

[20] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "ImageNet classification with deep convolutional neural networks," *Communications of the ACM* **60**(6), 84–90 (2017).

[21] Wu, Z., Shi, G., Chen, Y., Shi, F., Chen, X., Coatrieux, G., Yang, J., Luo, L., and Li, S., "Coarse-to-fine classification for diabetic retinopathy grading using convolutional neural network," *Artificial Intelligence in Medicine* **108**, 101936 (2020).

[22] Cannici, M., Ciccone, M., Romanoni, A., and Matteucci, M., "Asynchronous Convolutional Networks for Object Detection in Neuromorphic Cameras," in [*IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*], 1656–1665 (2019).

[23] Zhu, H., Wei, H., Li, B., Yuan, X., and Kehtarnavaz, N., "A Review of Video Object Detection: Datasets, Metrics and Methods," *Applied Sciences* **10**(21), 7834 (2020).

[24] Pal, S. K., Pramanik, A., Maiti, J., and Mitra, P., "Deep learning in multi-object detection and tracking: State of the art," *Applied Intelligence* **51**(9), 6400–6429 (2021).

[25] Krull, A., Buchholz, T.-O., and Jug, F., "Noise2void - learning denoising from single noisy images," in [*IEEE Conference on Computer Vision and Pattern Recognition*], 2129–2137 (2019).

[26] Hopfield, J. J., "Neural networks and physical systems with emergent collective computational abilities.," *Proceedings of the National Academy of Sciences* **79**(8), 2554–2558 (1982).

[27] Taherkhani, A., Belatreche, A., Li, Y., Cosma, G., Maguire, L. P., and McGinnity, T., "A review of learning in biologically plausible spiking neural networks," *Neural Networks* **122**, 253–272 (2020).

[28] Hochreiter, S. and Schmidhuber, J., "Long Short-Term Memory," *Neural Computation* **9**(8), 1735–1780 (1997).

[29] Graves, A. and Schmidhuber, J., "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks* **18**(5-6), 602–610 (2005).

[30] Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y., "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," in [*Conference on Empirical Methods in Natural Language Processing (EMNLP)*], 1724–1734, Association for Computational Linguistics, Doha, Qatar (2014).

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

[32] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE* **86**(11), 2278–2324 (Nov./1998).