Graph-Based Object Classification for Neuromorphic Vision Sensing Supplementary Material

Yin Bi, Aaron Chadha, Alhabib Abbas, Eirina Bourtsoulatze and Yiannis Andreopoulos
Department of Electronic & Electrical Engineering
University College London, London, U.K.
{ yin.bi.16, aaron.chadha.14, alhabib.abbas.13, e.bourtsoulatze, i.andreopoulos}@ucl.ac.uk

In this supplementary material, we explore how performance and complexity is affected when varying the key parameters of our approach. Via the ablation studies reported here, we justify the choice of parameters used for our experiments in the paper.

With regards to the parameters of the proposed graph CNNs, we experiment with the non-residual (i.e., plain) graph architecture (G-CNN) as a representative example, and explore the performance when varying the depth of graph convolution layer and the kernel size of graph convolution. Concerning the graph construction, our studied parameters are the time interval under which we extract events, the event sample size and the radius distance (R) used to define the connectivity of the nodes. All experiments reported in this supplementary note were conducted on the N-Caltech101 dataset, since it has the highest number of classes among all datasets. Finally, training methods and data augmentation follow the description given in Section 5.1 of the paper.

1. Event Sample Size for Graph Construction

The primary source of input compression is the non-uniform sampling of the events prior to graph-construction, which is parameterized by k in the paper. We explore the effects of this input compression by varying k and evaluating the accuracy to complexity (GFLOPs) tradeoff in Table 1. No compression (i.e., k = 1) gives accuracy/GFLOPs = 0.636/3.74, whereas increasing compression with k = 12 gives accuracy/GFLOPs = 0.612/0.26 (i.e., 93% complexity saving). This suggests that the accuracy is relatively insensitive to compression up to k = 12 (with k = 8 providing an optimal point) and it is the graph CNN that provides for state-of-the-art accuracy.

Table 1: Top-1 accuracy and complexity (GFLOPs) w.r.t. event sample size, parameterized by k.

k	Accuracy	GFLOPs
1	0.636	3.74
8	0.630	0.39
12	0.612	0.26

2. Radius Distance

When constructing graphs, the radius-neighborhood-graph strategy is used to define the connectivity of nodes. The radius distance (R) is an important graph parameter: when the radius is large, the number of generated graph edges increases, i.e., the graph becomes denser and needs increased GFLOPs for the convolutional operations. On the other hand, if we set a small radius, the connectivity of nodes may decrease to the point that it does not represent the true spatio-temporal relations of events, which will harm the classification accuracy. In this ablation study, we varied the radius distance to $\mathrm{R} = \{1.5, 3, 4.5, 6\}$, to find the best distance with respect to accuracy and complexity. The results are shown in Table 2, where we demonstrate that radius distance above 3 cannot improve the model performance while incurring significantly increased complexity. Therefore, in our paper we set the radius distance to 3. Note that when radius distance changes from 4.5 to 6, the required computation increases only slightly because of the maximum connectivity degree $\mathrm{D}_{\max}$ that is set to 32 to constrain the edge volume of graph.

Table 2: Top-1 accuracy and complexity (GFLOPs) w.r.t. radius distance

Radius distance	Accuracy	GFLOPs
1.5	0.551	0.33
3	0.630	0.39
4.5	0.626	0.98
6	0.624	1.19

3. Time Interval of Events

For each sample, events within a fixed time interval are randomly extracted to input to our object classification

framework. In this study, we test under various time intervals, i.e., 10, 30, 50 and 70 milliseconds, to see their effect on the accuracy and computation. The results are shown in Table 3. When extracting 30ms-events from one sample, the model achieves the highest accuracy, with modest increase in complexity over 10ms-events. Therefore, we opted for this setting in our paper.

Table 3: Top-1 accuracy and complexity (GFLOPs) w.r.t. the length of extracted events

length (ms)	Accuracy	GFLOPs
10	0.528	0.31
30	0.630	0.39
50	0.613	0.92
70	0.625	1.27

4. Depth of Graph Convolution Layers

As to the architecture of graph convolution networks, experimental studies by Li *et al.* [2] show that the model performance saturates or even drops when increasing the number of layers beyond a certain point, since graph convolution essentially pushes representations of adjacent nodes closer to each other. Therefore, the choice of depth of graph convolution layers (D) affects the model performance as well as its size and its complexity. In the following experiment, we tested various depths from 2 to 6, each followed by a max pooling layer, and subsequently concluding the architecture with two fully connected layers. The number of output channels ($\mathrm{C_{out}}$) in each convolution layer and the cluster size ($[\mathrm{s_h}, \mathrm{s_w}]$) in each pooling layers were as follows: *(i)* $\mathrm{D} = 2$: $\mathrm{C_{out}} = (128, 256)$, $[s_h, s_w] = (16 \times 12, 60 \times 45)$; *(ii)* $\mathrm{D} = 3$: $\mathrm{C_{out}} = (64, 128, 256)$, $[s_h, s_w] = (8 \times 6, 16 \times 12, 60 \times 45)$; *(iii)* $\mathrm{D} = 4$: $\mathrm{C_{out}} = (64, 128, 256, 512)$, $[s_h, s_w] = (4 \times 3, 16 \times 12, 30 \times 23, 60 \times 45)$; *(iv)* $\mathrm{D} = 5$: $\mathrm{C_{out}} = (64, 128, 256, 512, 512)$, $[s_h, s_w] = (4 \times 3, 8 \times 6, 16 \times 12, 30 \times 23, 60 \times 45)$; *(v)* $\mathrm{D} = 6$: $\mathrm{C_{out}} = (64, 128, 256, 512, 512, 512)$, $[s_h, s_w] = (2 \times 2, 4 \times 3, 8 \times 6, 16 \times 12, 30 \times 23, 60 \times 45)$. For all cases, the number of output channels of the two fully connected layers were 1024 and 101 respectively. The results are show in Table 4: while the highest accuracy is obtained when the depth is 5, complexity (GFLOPs) and size (MB) of the network is substantially increased in comparison to $\mathrm{D} = 4$. Therefore, in our paper, we set the depth of graph convolution layer to $\mathrm{D} = 4$.

5. Kernel Size

Kernel size determines how many neighboring nodes' features are aggregated into the output node. This comprises a tradeoff between model size and accuracy. Unlike conventional convolution, the number of FLOPs needed is independent of the kernel size. This is due to the local support property of the B-spline basis functions [1]. Therefore we only report the accuracy and model size with respect to various kernel sizes. In this comparison, the architecture is the same as the G-CNNs in Section 5.1, with the only difference being that the kernel size is increasing between 2 to 6. The results are shown in the Table 5. When kernel size is set as 3, 4, 5 and 6, the networks achieve the comparable accuracy, while the size of network increases significantly when the kernel size increases. In our paper, we set kernel size in the graph convolution to 5, due to the slightly higher accuracy it achieves. It is important to note that, even with a kernel size of 5 that incurs a larger-size model in comparison to size of 3, our approach is still substantially less complex than conventional deep CNNs, as shown in Table 3 in our paper.

Table 4: Top-1 accuracy, complexity (GFLOPs) and size (MB) of networks w.r.t. depth of convolution layer.

Depth	Accuracy	GFLOPs	Size (MB)
2	0.514	0.11	5.53
3	0.587	0.16	6.31
4	0.630	0.39	18.81
5	0.634	1.05	43.81
6	0.615	2.99	68.81

Table 5: Top-1 accuracy and size (MB) of networks w.r.t. kernel size

Kernel size	Accuracy	Size (MB)
2	0.543	5.02
3	0.626	8.30
4	0.621	12.90
5	0.630	18.81
6	0.627	26.02

6. Input Size for Deep CNNs

We investigate how the input size controls the tradeoff between accuracy and complexity for conventional deep CNNs trained on event images. We follow the training protocol and event image construction described in Section 5.2 of the paper, but now downsize the event image inputs to various resolutions prior to processing with the reference networks. The accuracy and complexity (GFLOPs) is reported on N-Caltech101 in Table 6. ResNet-50 offers the highest accuracy/GFLOPs tradeoff for conventional CNNs, ranging from 0.637/3.87 to 0.517/0.28. However, our RG-CNN trained on graph inputs surpasses accuracy of ResNet-50 for all resolutions, whilst offering comparable complexity (0.79 GFLOPs).

Table 6: Accuracy/GFLOPs of networks w.r.t. input size on N-Caltech101, for conventional deep CNNs with event image inputs.

Input Size	VGG_19	Inception_V4	ResNet_50
224×224	0.549/19.63	0.578/9.24	0.637/3.87
112×112	0.457/4.93	0.4272/1.63	0.595/1.02
56×56	0.300/1.29	0.343/0.22	0.517/0.28
G-CNNs	0.630/0.39	RG-CNNs	0.657/0.79

References

[1] Matthias Fey, Jan Eric Lenssen, Frank Weichert, and Heinrich Müller. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 869–877, 2018. 2

[2] Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 2