Masked Event Modeling: Self-Supervised Pretraining for Event Cameras
– Supplementary Material –

Simon Klenk[1,2*] David Bonello[1*] Lukas Koestler[1,2*] Nikita Araslanov[1,2] Daniel Cremers[1,2]
[1] Technical University of Munich [2] Munich Center for Machine Learning
{simon.klenk, david.bonello, lukas.koestler, nikita.araslanov, cremers}@tum.de
* Equal contribution

A. Semantic Segmentation: More Examples

In Figs. 7 to 9, we show additional qualitative examples for the downstream task of semantic segmentation. As discussed in Sec. 4.2 of the main paper, MEM-pretrained models tend to exhibit improved segmentation of fine-grained scene structures, such as pedestrians and lamp poles.

B. Convergence Curve

In Fig. 10, we show that the convergence speed of finetuning is considerably higher for MEM than from ViT-from-scratch. Recall also that the top-1 accuracy of the finetuned model is substantially higher than that of the ViT-from-scratch baseline, *e.g.* +18.66% on N-Caltech101 in Tab. 2.

C. Self-Supervised Baselines

We experiment with another baseline by reconstructing frames from events using E2VID [74] to further verify our design choices. We analyze the finetuning of both MAE and MEM on top of E2VID reconstructions in Tab. 7. Both methods are strong baselines, but directly using MEM on the raw event histograms is more effective. Moreover, the problem with a two-stage E2VID pipeline is that it uses significantly more compute and storage. Another serious disadvantage is that the classifier uses *reconstructed* images – a lossy transformation of the raw event data, which limits the accuracy and requires domain-specific labeled training data. Analogous to Tab. 6 in the main paper, we investigate two versions of MAE for completeness: E2VID+MAE-entire-hist, where the MAE loss is applied to the entire reconstructed histogram (our proposed modification for event histograms); and E2VID+MAE-only-mask, where the MAE loss is only applied to the masked patches (*cf*. [67]). While our modified version yields a significant improvement when pretraining on raw event histograms (*cf*. Tab. 6), it seems to be slightly beneficial to adopt the original formulation for reconstructed frames.

Method	N-Caltech101		N-Cars	
	FT	LP	FT	LP
E2VID+MEM	76.86	56.53	**94.53**	**90.88**
E2VID+MAE-entire-hist	77.45	58.70	93.09	88.33
E2VID+MAE-only-mask	**78.56**	**60.22**	94.39	89.51
MEM (ours)	**85.60**	**71.20**	**98.55**	**97.58**

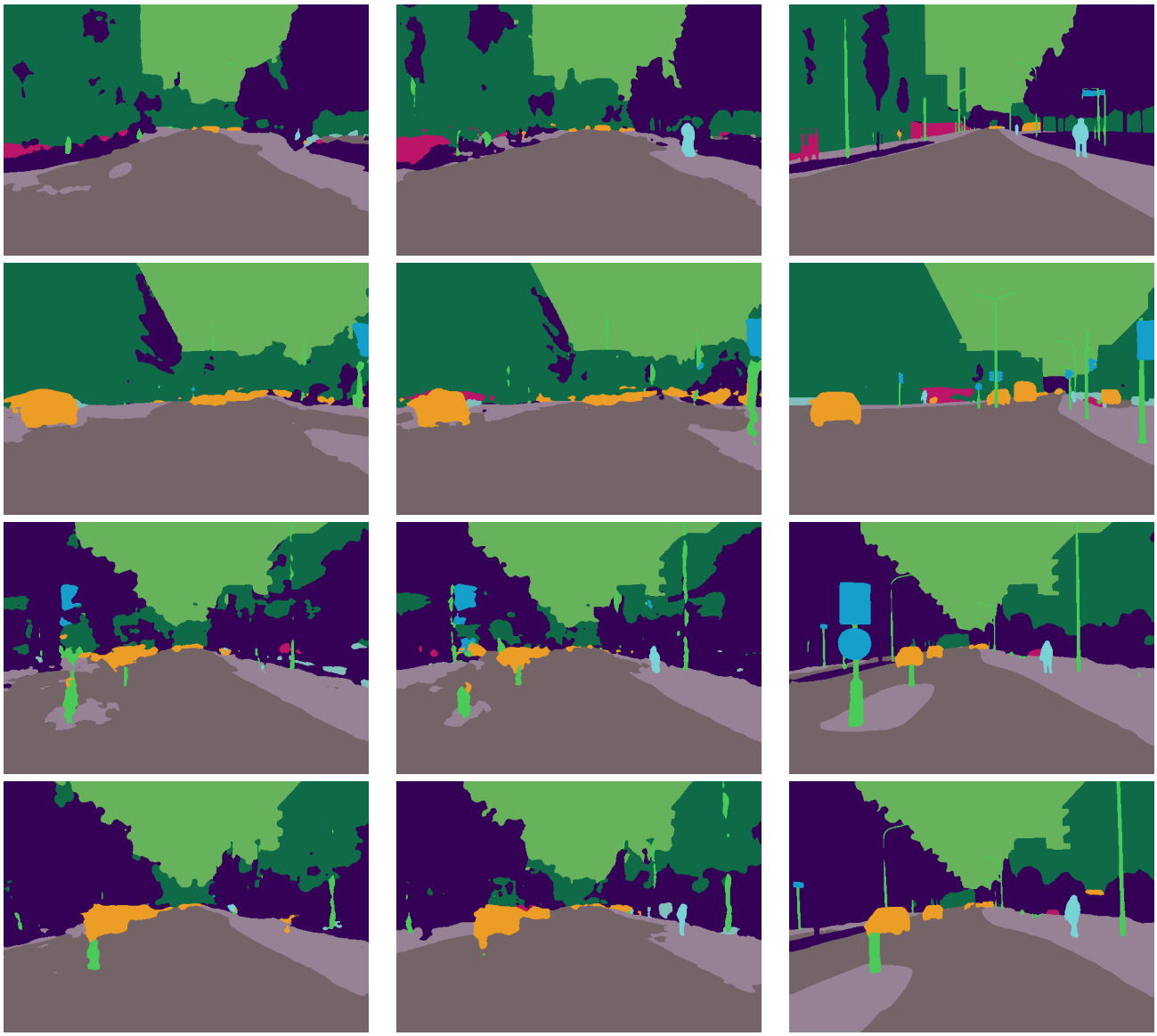
Table 7. E2VID baselines. Top-1 accuracy for finetuning (FT) and linear probing (LP) on N-Caltech101 and N-Cars.

Method	N-Caltech101	N-Cars
MEM + LP	**71.20**	**97.58**
MEM-NImNet + LP	68.19	89.82
Random Init + LP	25.34	75.94
ViT-from-scratch	66.94	92.71

Table 8. Top-1 accuracy of linear probing (LP) for MEM pretraining on N-Caltech101 and N-Cars. Random Init + LP is the LP accuracy of a randomly initialized ViT.

D. Linear probing (LP)

We also evaluate linear probing results with MEM pretraining. The conclusions from the main text still hold for LP: MEM-pretrained models achieve higher top-1 accuracy than ViT-from-scratch. We report top-1 LP accuracy on N-Caltech101 and N-Cars in Tab. 8. Compared to finetuning, LP benefits more from longer pretraining (*cf*. [67, Fig. 7]). We trained our model for 75 epochs on N-ImageNet, which is sufficient for standard finetuning, but proves to be too short for LP. Additionally, LP requires careful hyperparameter tuning (*cf*. [67, A.1 and Tab. 10]). While we used here the same hyperparameters for MEM as in finetuning, we could substantially improve upon the initial LP results by using new hyperparameters and removing regularization, following the insights of MAEs [67].



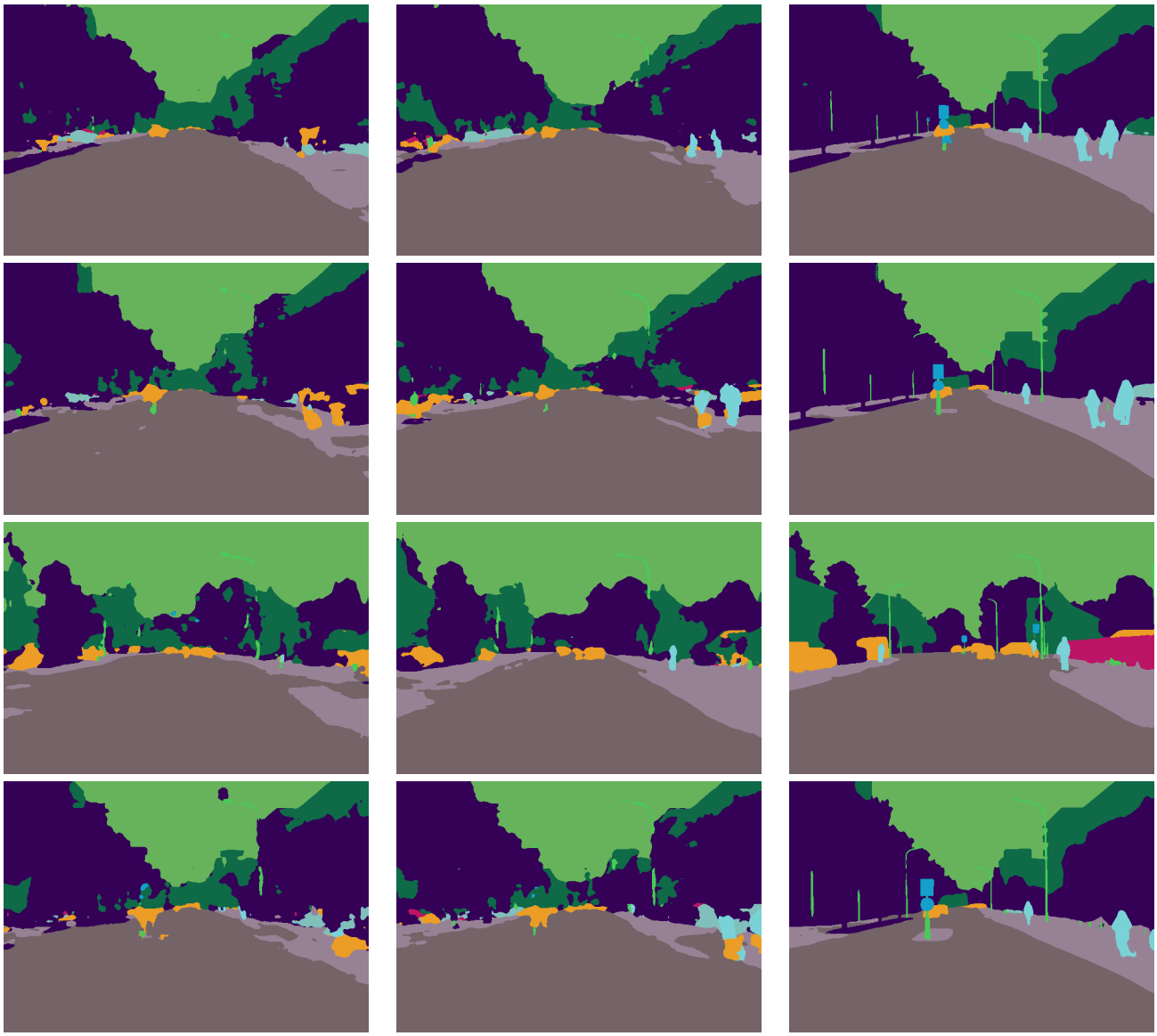
(a) ViT-from scratch (b) MEM (ours) (c) Ground truth

Figure 7. Semantic segmentation examples: (a) ViT-from-scratch, (b) MEM (ours), (c) the ground truth. MEM recovers the pedestrians (the right image half, 1st, 3rd, and 4th row), as well as the lamp pole (the right image half, 2nd, 3rd, and 4th row) more reliably.

E. Ablation of Input Representation

Event histograms already encode temporal information implicitly since they consider up to N_{max} events per histogram and accumulate the polarities into separate channels. As discussed in Sec. 4.6, we did not observe a significant benefit by explicitly including temporal information in the input, such as the event timestamps. In fact, as Tab. 9 shows, including this information into the event histograms as a third channel leads to slightly worse top-1 classification accuracy – compare lines *(i)* and *(ii)*.

To further investigate the importance of the temporal dimension for classification, we employ an 8-channel input histogram as the input. In contrast to our default 2-channel histograms, which collapse all recent events into two channels, this 8-channel histogram distributes the stream of N_{max} events into four equally spaced chunks of time and computes the histogram per chunk (similar to voxel grids [74]). Compared to simply using the latest timestamp in the third channel, this representation encodes the temporal information in more fine-grained manner. However, it does not yield a consistent advantage over our 2-channel baseline, as shown in lines *(iii)* and *(iv)* of Tab. 9. While the 8-channel representation somewhat improves the top-1 accuracy on N-



(a) ViT-from scratch (b) MEM (ours) (c) Ground truth

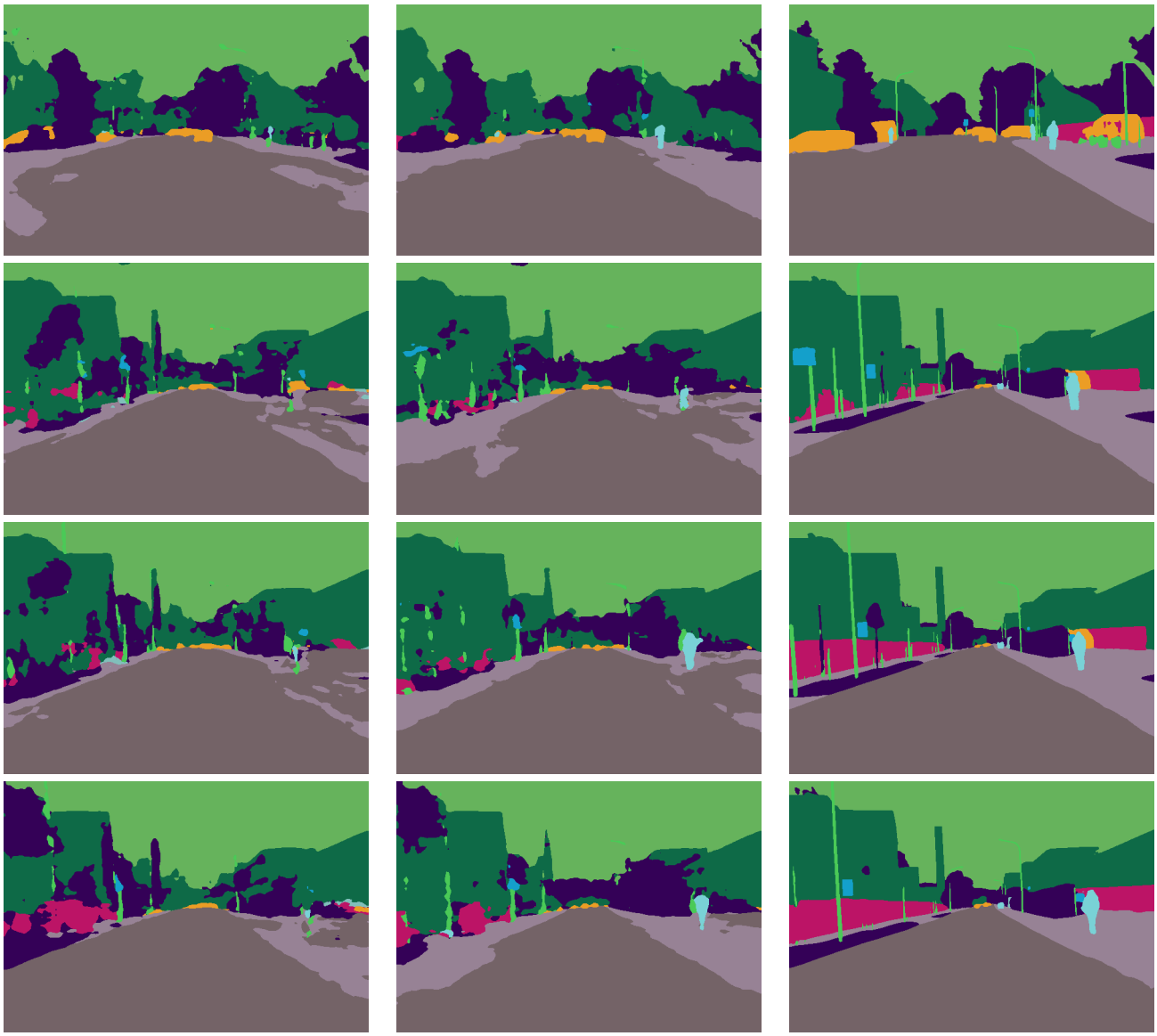
Figure 8. Semantic segmentation examples: (a) ViT-from-scratch, (b) MEM (ours), (c) the ground truth. MEM recovers the pedestrians (the right image half, 1st - 4th row), as well as the lamp pole (the right image half, 2nd and 4th row) more reliably.

Cars, the result is considerably worse on N-Caltech101.

The marginal benefit of the more explicit temporal encoding, as demonstrated in these experiments, has an intuitive explanation if we consider the underlying task – object classification. Semantic meaning is easily accessible from the *spatial* context, such as object shape, rather than the temporal distribution of brightness changes, which the more explicit temporal representations provide.

F. Reconstruction and Token Visualization

We visualize additional codebook vectors in Fig. 11. We observe that codebook vectors, which have a fixed index, tend to exhibit recurring shape characteristics and a consistent preference for polarity (*e.g.*, either positive or negative or an equal share of both). The most common codebook vectors are completely blank since the event histogram is sparsely populated with event count values. Due to redundancy, we do not visualize these blank codebook vectors. Although all codebook vectors are fixed, the decoder adapts each patch to its surroundings to form a coherent im-



(a) ViT-from scratch (b) MEM (ours) (c) Ground truth

Figure 9. Semantic segmentation examples: (a) ViT-from-scratch, (b) MEM (ours), (c) ground-truth annotation map. MEM pretraining recovers the pedestrians (the right image half, 1st - 4th row), as well as the lamp pole (the left image half, 2nd - 4th row) more reliably.

age. Hence, the visualized examples of each decoded codebook vector show some appearance variation (column-wise in Fig. 11). All visualized codebook vectors are rendered using the test set of N-Cars [47].

Complementing Fig. 4, Fig. 12 illustrates the reconstruction of masked patches during pretraining on all datasets used in this work. As we discuss in Sec. 4.5, the MAE pretraining task can also be employed on event histograms (in contrast to the original MAE paper [67]). However, it requires the loss to be formulated on the entire histogram. We found the reconstructions from eMAE, which are visualized in Fig. 13, to be not as sharp as for MEM.

G. Implementation Details

As discussed in Sec. 4, our MEM pretraining, as well as the baselines ViT-from-scratch, ViT-1k, and ViT-21k, share the same implementation. As detailed next, we make a significant effort to ensure strong baseline performance by using best-practice training techniques.

ViT Architecture We use ViT-Base described in [14] with patch size 16×16. It consists of 12 layers and has 12

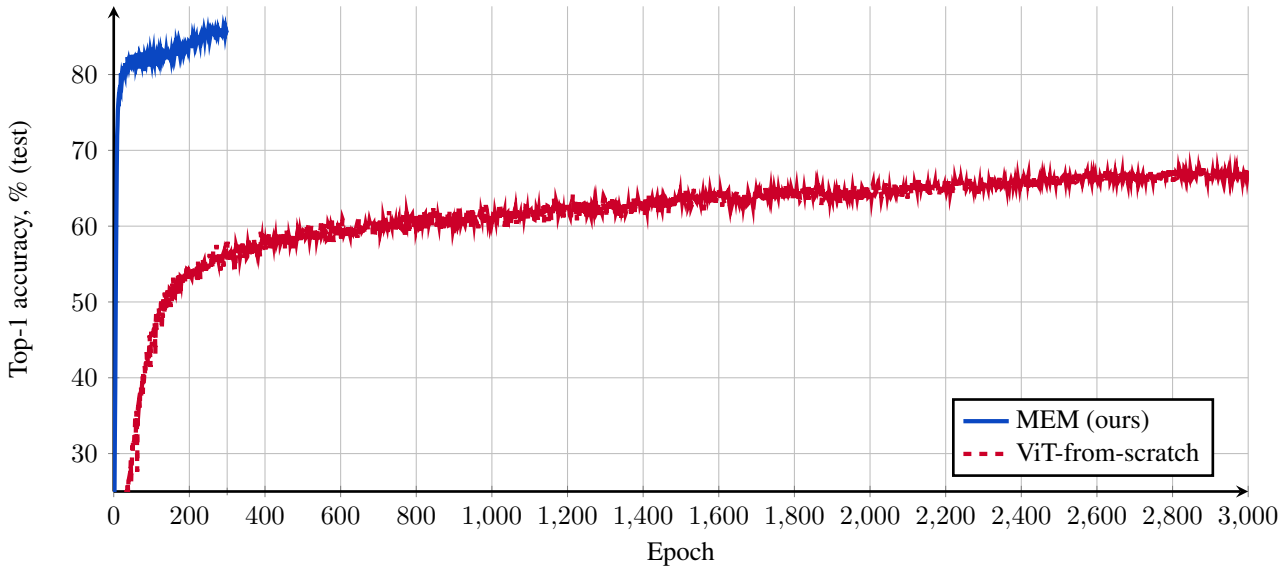


Figure 10. Finetuning accuracy vs. epochs on N-Caltech101 [47]. With our proposed pretraining (MEM), the accuracy increases much faster. It reaches a higher final accuracy of 85.60% compared to finetuning without pretraining (ViT-from-scratch), where the final accuracy is only 66.94%. Both the pretraining and the finetuning tasks use the entire N-Caltech101 train dataset.

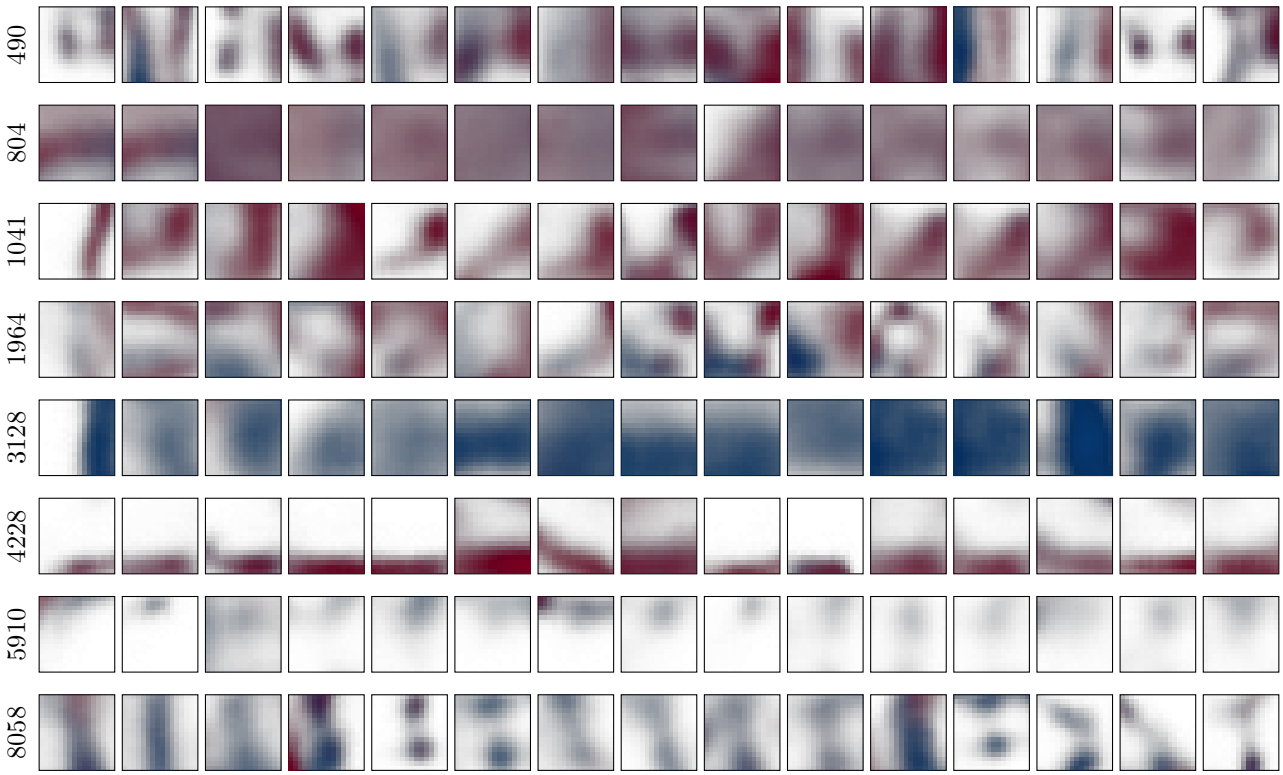


Figure 11. Additional examples of decoded codebook vectors with the codebook indexes of 490, 804, 1041, 1964, 3128, 4428, 5910, and 8058. Although all codebook vectors are fixed, the decoder adapts each patch to its surroundings to form a coherent image. Observe that each codebook index corresponds to a specific visual feature, *e.g.*, a red horizontal line at the bottom of the patch (4228) or a round mixture of red and blue polarity (1964). The codebook size is 8092. These visualizations are rendered from the test set of N-Cars [47].

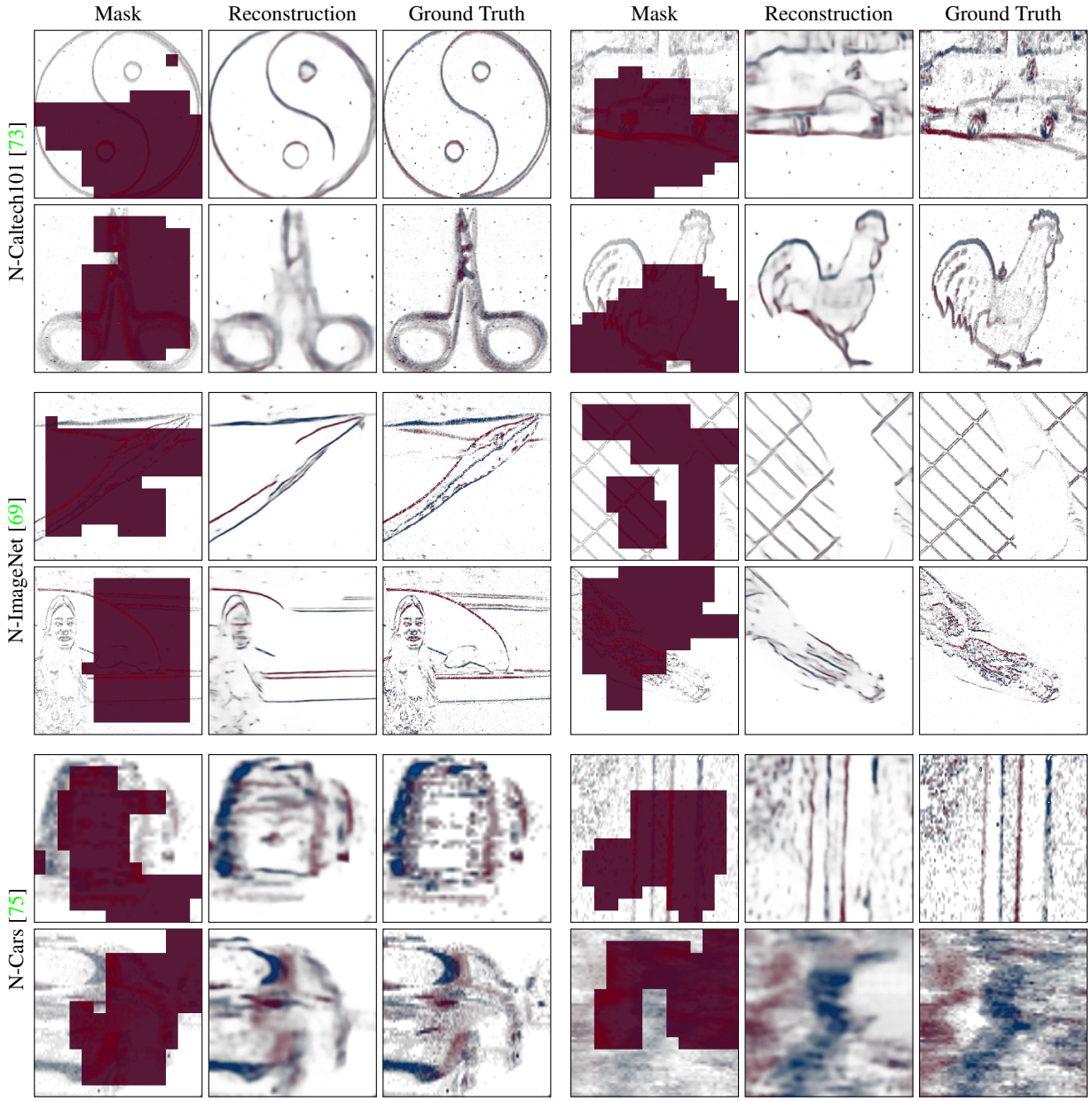


Figure 12. Additional masked patch predictions. From left to right: We visualize the masked input histograms, the reconstructions during pretraining, and the ground truth for N-Caltech101 [37] (top), for N-Imagenet [28] (middle) and for N-Cars [47] (bottom). We observe that the ground truth can be recovered even if large parts of the input to the ViT are masked. The ViT predicts the tokens for all masked patches, which are then decoded into the predicted event histogram by the decoder of the dVAE. These visualizations are rendered from the test set.

heads in the self-attention layers. The feature size is 768, maintained by MLPs with 3072 hidden units. We add a linear projection on the ViT features during pretraining, which outputs the visual tokens. We discard this linear projection during finetuning and train a new linear layer for classification. We employ relative positional encoding [14].

Hyperparameters We report hyperparameters for dVAE in Tab. 11, pretraining in Tab. 12, and finetuning in Tab. 13. As discussed in Sec. 3.2, gradient clipping is a vital hyperparameter. In Tab. 10, it can be seen that without gradient clipping, the accuracy on N-Caltech101 and N-Cars is worse than the baseline ViT-from-scratch. Hence, gradient clipping is essential to employ MEM pretraining on sparse event histograms successfully. Note that gradient clipping

Ablation	N-Caltech101	N-Cars
(i) MEM	**85.60**	**98.55**
(ii) w/ timestamps (3rd chan.)	84.90	98.10
with 33% pretrain steps:		
(iii) MEM (2 channels)	**81.17**	95.16
(iv) 8-channels	79.70	**95.84**

Table 9. A study of alternative input representations on N-Caltech101 [73] and N-Cars [75]. As the baseline, here we use 2-channel histograms (see Sec. 4.6) of size 224×224, patch size 16, masking ratio 50%, RandAugment [66] and gradient clipping (see Tabs. 11 and 12 for other hyperparameters).

Ablation	N-Caltech101	N-Cars
(i) MEM	**85.60**	**98.55**
(ii) ViT-from-scratch	66.94	92.71
(iii) MEM w/o grad. clip.	22.87	90.73

Table 10. Ablation of gradient clipping for dVAE and pretraining stage on N-Caltech101 and N-Cars. Gradient clip values are in Tab. 11 and Tab. 12. Gradient clipping is essential to employ MEM pretraining on sparse event histograms. Note that gradient clipping is not employed in the RGB setting [65, 67].

is not used in the RGB setting [65, 67].

G.1. Details on Event Preprocessing

After loading all events for a given sample (*e.g.*, accumulated over 300ms in N-Caltech101), we slice the events in time by randomly selecting one contiguous batch comprising up to $N_{\text{max}} = 30,000$ events. During training, we perform *(i)* a random flip of all event polarities (with probability $p = 0.5$); *(ii)* a random horizontal flip (with probability $p = 0.5$); and *(iii)* random shifts (per event) by Δx and Δy of x-coordinates and y-coordinates using uniform sampling, *i.e.* $\Delta x \sim \mathcal{U}(-15, 15)$ and $\Delta y \sim \mathcal{U}(-15, 15)$.

We accumulate the augmented events into a two-channel histogram. For N-Caltech101 and N-Cars, we resize the histograms to spatial resolution 224×224. For N-ImageNet, we resize the histogram to 256×341 and randomly crop the image to 224×224. Next, we remove "hot pixels", a variant of noise specific to event cameras, which manifests as a continuously triggering event [68]. We define a pixel as a "hot pixel" if its event count is ten standard deviations above the mean value in the event batch. We normalize the histogram values to $[0, 1]$. Lastly, during training, we perform RandAugment [10] with two operations and a magnitude of 20. The three stages of MEM (dVAE, pretraining, and finetuning) share the same input preprocessing (*i.e.*, the event histogram).

G.2. Datasets

We use the official train and test splits of N-ImageNet [28] and N-Cars [47] and DSEC-Semantic [19]. For N-Caltech101 [37], we randomly split the data into 80% training data and 20% test data. We ran 5-fold cross-validation to confirm that all random splits yield approximately the same result on N-Caltech101. The top-1 accuracy on the test sets in these experiments were 84.6%, 84.7%, 85.3%, 85.6% (reported in the main text), and 85.8%.

References

[65] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers. In *ICLR*, 2022. 7

[66] Ekin Dogus Cubuk, Barret Zoph, Jonathon Shlens, and Quoc Le. RandAugment: Practical automated data augmentation with a reduced search space. In *NeurIPS*, 2020. 7

[67] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022. 1, 4, 7, 9

[68] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbruck. v2e: From video frames to realistic DVS events. In *CVPR*, pages 1312–1321, 2021. 7

[69] Junho Kim, Jaehyeok Bae, Gangin Park, Dongsu Zhang, and Young Min Kim. N-ImageNet: Towards robust, fine-grained object recognition with event cameras. In *CVPR*, pages 2146–2156, 2021. 6

[70] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8

[71] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 8

[72] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 8

[73] Garrick Orchard, Ajinkya Jayawant, Gregory K. Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in neuroscience*, 9:437, 2015. 6, 7, 9

[74] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *CVPR*, pages 3857–3866, 2019. 1, 2

[75] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. HATS: Histograms of averaged time surfaces for robust event-based object classification. In *CVPR*, pages 1731–1740, 2018. 6, 7, 9

Hyperparameter	N-ImageNet [28]	N-Caltech101 [37]	N-Cars [47]
Optimizer	Adam [70]	Adam [70]	Adam [70]
Optimizer momentum	$\beta_1, \beta_2 = (0.9, 0.999)$	$\beta_1, \beta_2 = (0.9, 0.999)$	$\beta_1, \beta_2 = (0.9, 0.999)$
Learning rate	1e-3	2e-4	2e-4
Learning rate schedule	exponential (0.99)	exponential (0.99)	exponential (0.99)
Learning rate layer decay	0.98	0.98	0.98
KL weight	1e-10	1e-10	1e-10
Batch size	512	192	192
Grad clip	1e-2	1e-2	1e-2
Epochs	50	300	300

Table 11. Hyperparameters for the dVAE.

Hyperparameter	N-ImageNet [28]	N-Caltech101 [37]	N-Cars [47]
Optimizer	AdamW [72]	AdamW [72]	AdamW [72]
Optimizer momentum	$\beta_1, \beta_2 = (0.9, 0.95)$	$\beta_1, \beta_2 = (0.9, 0.95)$	$\beta_1, \beta_2 = (0.9, 0.95)$
Learning rate	1e-4	5e-4	3e-4
Learning rate schedule	cosine decay [71]	cosine decay [71]	cosine decay [71]
Warmup steps	1000	1000	1000
Weight decay	0.05	0.05	0.05
Batch size	512	512	384
Grad clip	30	30	30
Epochs	75†	3000	1000‡

Table 12. Hyperparameters for pretraining. †Cosine scheduler set for 300 epochs, but for computational reasons, only training for 75 epochs. ‡ Cosine scheduler set for 3000 epochs, but only training for 1000 epochs.

Hyperparameter	N-ImageNet [28]	N-Caltech101 [37]	N-Cars [47]
Optimizer	AdamW [72]	AdamW [72]	AdamW [72]
Optimizer momentum	$\beta_1, \beta_2 = (0.9, 0.95)$	$\beta_1, \beta_2 = (0.9, 0.95)$	$\beta_1, \beta_2 = (0.9, 0.95)$
Learning rate	1e-3	4e-3	5e-4
Learning rate schedule	cosine decay [71]	cosine decay [71]	cosine decay [71]
Learning rate layer decay	0.65	0.65	0.65
Warmup epochs	20	20	20
Weight decay	0.3	0.05	0.05
Drop path	0.1	0.1	0.1
Dropout	0.0	0.1	0.1
Batch size	1024	1024	1024
Epochs	200†	300	300

Table 13. Hyperparameters for finetuning. †Cosine scheduler set for 300 epochs, but for computational reasons, only finetuning for 200 epochs. We report the exponential moving average accuracy on N-Imagenet with a decay factor of 0.9999.

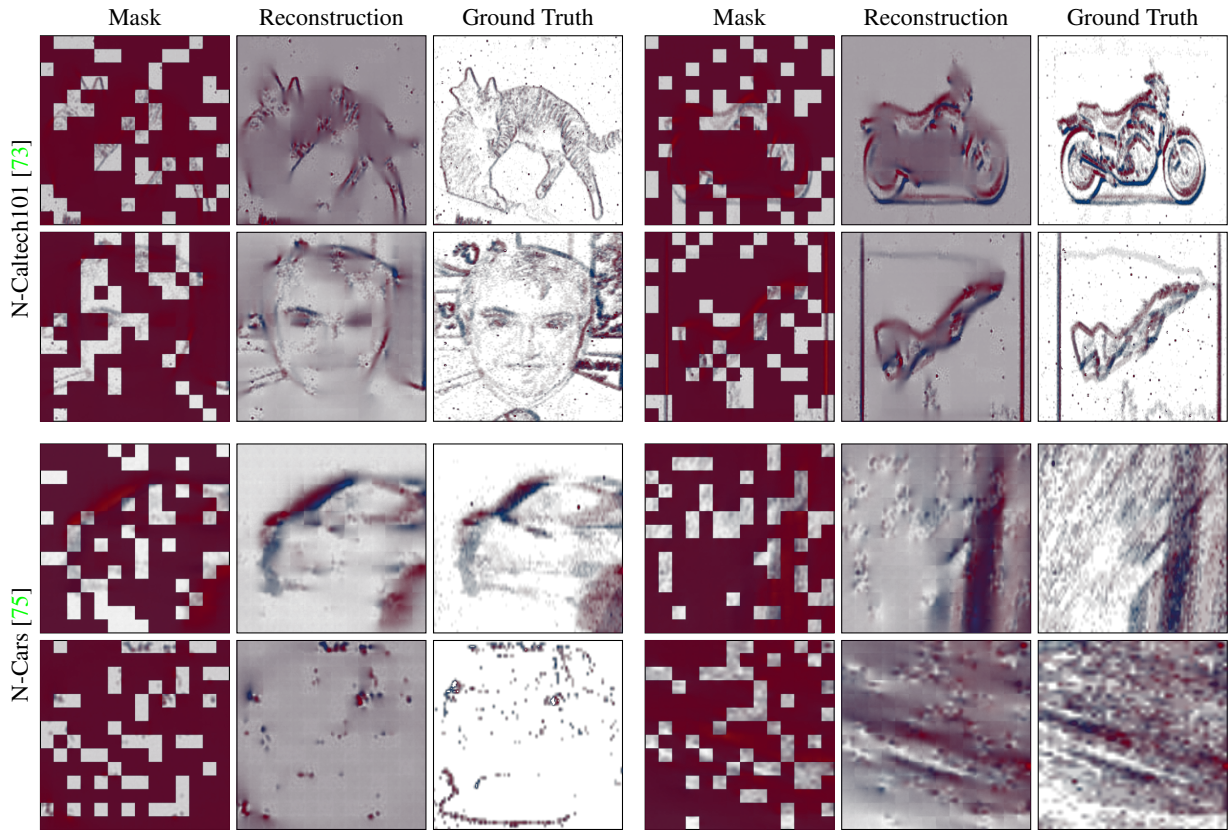


Figure 13. Additional masked patch predictions with the MAE loss (eMAE-entire-hist, see Tab. 6). From left to right: We visualize the masked input histograms, the reconstructions during eMAE-entire-hist pretraining, and the ground truth for N-Caltech101 [37] (top) and for N-Cars [47] (bottom). The MAE pretraining tasks struggles to recover sharp reconstructions, compared to MEM (*cf*. Fig. 12). We employ the default values of the MAE paper [67], *i.e*. a masking ratio of 75%, and random masking.