A. Details on Pretrained Tokenizers

In this paper, we use the visual tokenizer of a pretrained image VQ-VAE from [2, 6], which is also called discrete VAE [6]. For DALL-E [6], the tokenizer of the VQ-VAE is trained to transform each 256×256 image into a 32×32 image token map according to a visual codebook, while the decoder of the VQ-VAE is trained to reconstruct each input image from its tokens. The vocabulary size of the visual tokens is 8192.

B. The Influence of Loss Weight λ

In our experiments, we simply set loss weight $\lambda = 1$. We also test its sensitivity on three video recognition datasets. As shown in Table 1, the performance on all downstream tasks is not sensitive to λ.

λ	SSv2	Diving48	K400
0.5	70.7	86.3	80.6
1.0	70.6	86.7	80.6
2.0	70.7	86.4	80.7

Table 1. Performance sensitivity of loss weight λ.

C. The Initialization of Video Stream

In BEVT, we initialize the video stream with the weights of IN-1k pretrained on the image stream before performing two-stream joint pretraining. Following the method in [4], we initialize the 3D patch embedding layer by duplicating the weights of 2D patch embedding layers and then multiplying the whole matrix by the reciprocal of temporal kernel size (*i.e.*, keeping the mean and variance of the output unchanged). The 3D relative positional bias is initialized by duplicating the 2D version. Since the weights of self-attention modules are irrelevant to the input sequence length, we initialize the remaining layer weights directly from the image transformer.

D. Implementation Details for Downstream Tasks

For downstream tasks, we finetune the pretrained Video Swin models for 60 epochs with a batch size of 64 and the clip length is 32. We use the AdamW [5] optimizer with a linear warm-up and a cosine learning rate schedule for both pretraining and finetuning.

E. Implementation Details for Pretraining and Finetuning TimeSformer

In the ablation study, we also instantiate the BEVT framework with TimeSformer [1]. In contrast to Video Swin Transformer, TimeSformer is extended from the ViT [3] architecture, so the length of the token sequence is not down-sampled. Therefore, we do not need any upsampling modules and directly employ a softmax-based linear classifier as the BEVT decoder. The pretraining setting of TimeSformer is the same as that of Video Swin. For downstream tasks, we finetune the pretrained TimeSformer for 30 epochs on K400 and Diving48, and 15 epochs on SSv2. For both pretraining and finetuning, the input clip length of TimeSformer is 8 (we do not use TimeSformer-L or TimeSformer-HR here). We use the AdamW [5] optimizer with a linear warm-up and a cosine learning rate schedule. During inference, following [1], we only use 3 views (3 spatial crops).

References

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, 2021. 1

[2] Xiaoyi Dong, Jianmin Bao, Ting Zhang, Dongdong Chen, Weiming Zhang, Lu Yuan, Dong Chen, Fang Wen, and Nenghai Yu. Peco: Perceptual codebook for bert pre-training of vision transformers. *arXiv preprint arXiv:2111.12710*, 2021. 1

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1

[4] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. *arXiv preprint arXiv:2106.13230*, 2021. 1

[5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2018. 1

[6] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 1