Supplementary Material

A Q&A Session

Where does the performance gain come from? Is it driven by the diffusion model only? The performance gain in our model is not solely driven by the diffusion model. It is, in fact, a result of the combined efforts of two major stages: the fMRI encoder and the stable diffusion model.

In the first stage, the fMRI encoder plays a crucial role in learning representations from the brain. It effectively captures the complex spatiotemporal information embedded in the fMRI data, allowing the model to understand and interpret the underlying neural activities. This step is particularly important as it forms the foundation of our model and significantly influences the subsequent steps.

In the second stage, the stable diffusion model steps in to generate videos. One of the key advantages of our stable diffusion model over other generative models, such as GANs, lies in its ability to produce higher-quality videos. It leverages the representations learned by the fMRI encoder and utilizes its unique diffusion process to generate videos that are not only of superior quality but also better align with the original neural activities.

What does the fMRI encoder learn? Why don't we train an fMRI to object classifier, followed by a generation model? The fMRI encoder is designed to learn intricate representations from brain activity. These representations go beyond simple categorical information to encompass more nuanced semantic details that can't be adequately captured by discrete class labels (e.g. image texture, depth, etc.). Due to the richness and diversity of human thought and perception, a model that can handle continuous semantics, rather than discrete ones, is necessary.

The proposal to train an fMRI-to-object classifier followed by a generation model does not align with our goal of comprehensive brain decoding. This is largely due to a crucial trade-off between classification complexity and solution space:

- **Classification Complexity:** Classifying fMRI data into a large number of classes (e.g., 1000 classes) is non-trivial. As reported in [2], reasonable performance can only be achieved in a smaller classification task (less than 50-way), due to the limited data per category and the complexity of the task.
- **Limited Solution Space:** The solution space of discrete classes is significantly more restricted than that of continuous semantics. Thus, a classifier may not capture the complex, multi-faceted nature of brain activities.

This trade-off illustrates why a classifier might not be the best approach for this task. In contrast, our proposed method focuses on learning continuous semantic representations from the brain, which better reflects the complexity and diversity of neural processes. This approach not only improves the quality of the generated videos but also provides more meaningful and interpretable insights into brain decoding.

Is the fMRI-video generation pipeline simply imputing missing frames in a sequence of static images based on the fMRI-image generation pipeline? No, the fMRI-to-video generation process is not merely an imputation on the fMRI-to-image generation pipeline. While both involve generating visual content based on fMRI data, the tasks and their complexities are fundamentally different.

The fMRI-to-image generation involves mapping brain activity to a static image. This task primarily focuses on spatial information, that is, which brain regions are active and how they relate to elements in an image.

In contrast, the fMRI-to-video generation task involves mapping brain activity to dynamic videos. This task is considerably more complex as it requires the model to capture both spatial information and temporal dynamics. It's not just about predicting which brain regions are active, but also about understanding how these activations change over time and how they relate to moving elements in a video.

Adding to the complexity is the hemodynamic response inherent in fMRI data, which introduces a delay and blur in the timing of neural activity. This necessitates careful handling of the temporal aspects in the data. Furthermore, the temporal resolution of fMRI is quite low, making it challenging to capture fast-paced changes in neural activity.

We also use a stable diffusion process as our generative model, which is a probabilistic model. This means that the generation process involves a degree of randomness, leading to slight differences during each generation for video frames. Additionally, in video generation, we need to ensure consistency across video frames, which adds another layer of complexity.

B More Implementation Details

Large-Scale Pre-training The large-scale pre-training uses the same setup as the MBM described in [6]. A ViT-large-based model with a 1-dimensional patchifier is trained with hyperparameters shown in Tab. B.1. The training takes around 3 days using 8RTX3090 GPUs. The training is performed on the 600,000 fMRI from HCP. Same as the literature, after the large-scale pre-training, the autoencoder is tuned with fMRI data from the target dataset, namely, Wen (2018), using MBM as well. The tuning is performed using a small learning rate and epochs.

parameter	value	parameter	value	parameter	value	parameter	value
patch size	16	encoder depth	24	decoder embed dim	512	clip gradient	0.8
embedding dim	1024	encoder heads	16	learning rate	2.5e-4	weight decay	0.05
mask ratio	0.75	decoder depth	8	warm-up epochs	40	batch size	500
mlp ratio	1.0	decoder heads	16	epochs	500	optimizer	AdamW

Table B.1. Hyperparameters used in the large-scale pre-training

Multimodal Contrastive Learning In this step, we will take the pre-trained fMRI encoder from the previous step and augment it with temporal attention heads to accommodate multiple fMRI. Then contrastive learning is performed with fMRI-image-text triplets. The image is a randomly-picked frame from an fMRI scan window. As mentioned, there are two important factors in the contrastive: batch size and data augmentations. Therefore, data augmentations are applied for all modalities. Random sparsification is used for fMRI, where 20% of voxels are randomly set to zeros each time. The random crop is applied to videos with a probability of 0.5. To augment the frame captions, we apply synonym augmentation and random word swapping. Due to a small dataset size ($\sim$4000), we use a dropout rate of 0.6 to avoid overfitting. For Subject 1 and 2, the training is performed with a batch size of 20, while a batch size of 32 is used due to fewer fMRI voxels with Subject 3. Training for all subjects is performed for 1,2000 steps with a learning rate of 2×10^{-5}. The training takes around 10 hours using one RTX3090.

Training of Augmented Stable Diffusion The stable diffusion model is augmented with temporal attention heads for video generation. We train the augmented stable diffusion model with videos from the target datasets. The videos are downsampled from 30 FPS to 3 FPS at a resolution of 256×256 due to limited GPU memory, even though our model can work with the full time resolution. This step is important for two reasons: 1) the augmented temporal heads are untrained; 2) the stable diffusion is pre-trained at a resolution of 512×512, so we need to adapt it to a lower resolution.

During the training, we update the self-attention heads ("attn1"), cross-attention heads ("attn2"), and temporal attention heads ("attn_temp"). The training is performed with text conditioning for 800 steps. We use a learning rate 2×10^{-5} and a batch size of 14. The training takes around 2 hours using one RTX3090. Visual results show that videos of high quality can be generated with text conditioning after this step.

Co-training The fMRI encoder produces embeddings of dimensions 77×768, which are used to condition the augmented stable model during co-training. The whole fMRI encoder is updated, and only part of the stable diffusion is updated (same as the last step). The training is performed with a batch size of 9 and a learning rate of 3×10^{-5} for 1,5000 steps. The training takes around 16 hours using one RTX3090.

Inference All samples are generated with 200 diffusion steps using fMRI adversarial guidance. The fMRI adversarial guidance uses an average fMRI as the negative guidance with a guidance scale of 12.5.

C Analysis of Visual Results

We test on all three subjects in Wen (2018) dataset. Around 6000 voxels are identified as ROI for Subject 1 and 2, while around 3000 voxels are identified for Subject 3. Thus, a larger batch size can be used when training with Subject 3, which may be the reason for its better numeric evaluation results. Nonetheless, all three subjects show consistent generation results. Some samples are shown in Fig. C.1.

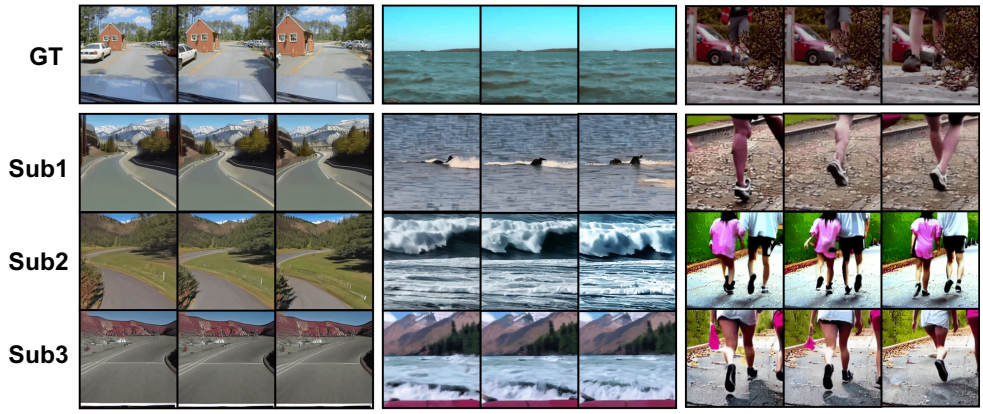

Figure C.1. Samples from different subjects.

Visual results of the ablation studies are shown in Fig. C.2. The Full model is trained with our full pipeline and inference with adversarial guidance. In contrastive learning ablation, we tested with incomplete modality, namely, image-fMRI and text-fMRI, respectively. Similar to the numeric evaluations, using incomplete contrastive gave an unsatisfactory visual result compared to using all three modality. However, incomplete modality still outperformed inferencing without adversarial guidance significantly, which generated visually meaningless results.

Figure C.2. Reconstruction samples for ablation studies. The Full model uses full modality contrastive learning with adversarial guidance.

Some fail cases are shown in Fig. C.3. It is observed that even though some fail cases generated different animals and objects compared to the groundtruth, other semantics like the motions, color, and scene dynamics can still be correctly reconstructed. For example, even though the airplane and flying bird are not reconstructed, similar fast-motion scenes are recovered in Fig. C.3.

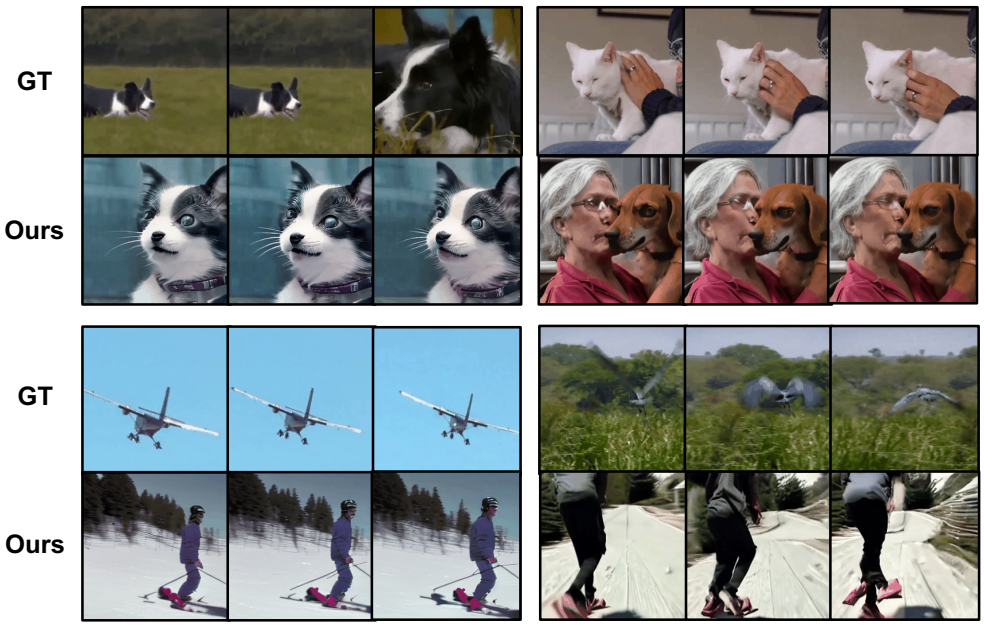

Figure C.3. Fail cases.