Exploring Behavior-Relevant and Disentangled Neural Dynamics with Generative Diffusion Models

Yule Wang
Georgia Institute of Technology
Atlanta, GA, 30332 USA
yulewang@gatech.edu

Chengrui Li
Georgia Institute of Technology
Atlanta, GA, 30332 USA
cnlichengrui@gatech.edu

Weihan Li
Georgia Institute of Technology
Atlanta, GA, 30332 USA
weihanli@gatech.edu

Anqi Wu
Georgia Institute of Technology
Atlanta, GA, 30332 USA
anqiwu@gatech.edu

Abstract

Understanding the neural basis of behavior is a fundamental goal in neuroscience. Current research in large-scale neuro-behavioral data analysis often relies on decoding models, which quantify behavioral information in neural data but lack details on behavior encoding. This raises an intriguing scientific question: "*how can we enable in-depth exploration of neural representations in behavioral tasks, revealing interpretable neural dynamics associated with behaviors*". However, addressing this issue is challenging due to the varied behavioral encoding across different brain regions and mixed selectivity at the population level. To tackle this limitation, our approach, named "BeNeDiff", first identifies a fine-grained and disentangled neural subspace using a behavior-informed latent variable model. It then employs state-of-the-art generative diffusion models to synthesize behavior videos that interpret the neural dynamics of each latent factor. We validate the method on multi-session datasets containing wide-field calcium imaging recordings across multiple brain regions of the dorsal cortex. By guiding the diffusion model to activate individual latent factors, we verify that the neural dynamics of latent factors in the disentangled neural subspace provide interpretable quantifications of the behaviors of interest across multiple brain regions. Meanwhile, the neural subspace in BeNeDiff demonstrates high disentanglement and neural reconstruction quality. Our codes are available at https://github.com/BRAINML-GT/BeNeDiff.

1 Introduction

Understanding and elucidating the complex interrelationships between behavioral data and neural population activity is a long-standing goal in systems neuroscience [Batty et al., 2019; Gomez-Marin et al., 2014; Krakauer et al., 2017; Berman, 2018]. Exploring the neural basis of behavior not only deepens our basic knowledge of brain functions but also establishes a foundation for developing improved treatments for psychiatric and neurological conditions [Vieira et al., 2017; Ibáñez et al., 2018]. Significant progress has been achieved in developing computational toolkits for neuro-behavioral decoding by using behavior video data [Whiteway et al., 2021; Batty et al., 2019; Musall et al., 2019; Stringer et al., 2019]. These methods perform region-based behavior decoding to map neural activity across multiple brain regions of the dorsal cortex to the behaviors from the videos. However, these methods only quantify how much behavioral information is encoded in neural populations, but do not reveal the details of such encoding. There has been markedly less focus,

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

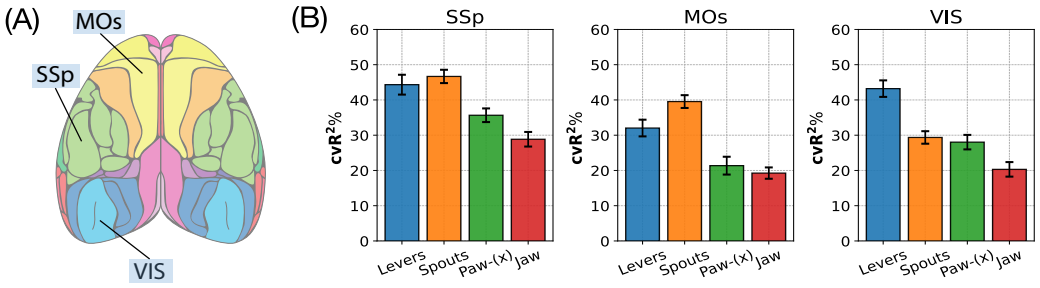

Figure 1: **Empirical study across multiple brain regions of dorsal cortex neural recordings of a mouse in a visual decision-making task**. **(A)** The Brain Atlas map [Lein et al., 2007]. **(B)** Neural signals in various brain regions (SSp, MOs, and VIS) exhibit mixed selectivity in behavior of interest decoding. "Levers", "Spouts", "Paw-(x)", and "Jaw" are four behaviors of interest. $\mathbf{cvR}^2$ is short for cross-validation coefficient of determination. The higher, the better.

with cortex-wide signals, on enabling in-depth exploration of neural activities during behavioral tasks, where specific neural patterns reveal dynamic evolutions corresponding to distinct behaviors of interest. However, empirically addressing this scientific question is challenging due to neural population activities in various brain regions exhibiting mixed selectivity [Sani et al., 2021; Hasnain et al., 2023], responding robustly to multiple behaviors of interest. We further verify this finding through an empirical study across three brain regions on the dorsal cortex of head-fixed mice [Musall et al., 2019] (shown in Figure 1).

To tackle this issue, we propose a method - Exploring **Be**havior-Relevant and Interpretable **Ne**ural Dynamics with Generative **Diff**usion Models - ("BeNeDiff"). We first employ a neural latent variable model (LVM) to identify orthogonal and disentangled neural latent subspace. This is achieved through a semi-supervised variational autoencoder, which integrates behavioral labels to rotate the subspace. Subsequently, our main idea is to explore the neural dynamics of each latent factor in the learned subspace for distinct quantifications of the behaviors of interest. However, such a workflow is non-trivial since naïve latent manipulation produces samples not conform to the original distribution, leading to mapped video-based behavioral data that loses its validity (we further detail this part in Method Section 3.2.1).

Notably, we aim to investigate the behavioral-specificity of neural latent factors in a generative fashion. We leverage state-of-the-art video diffusion models (VDMs) to generate behavior videos predicted to *activate* individual latent factors along the single-trial trajectory. Technically, the VDMs are capable of capturing the overall temporal dynamics and synthesizing behavior videos in a classifier-guided manner [Dhariwal and Nichol, 2021]. Inspired by Noise-Contrastive Estimation [Gutmann and Hyvärinen, 2010], the guidance objective is formulated to amplify the variance of the selected latent factor along its neural trajectory while suppressing the variance of the neural trajectories of the other latent factors.

We conduct experiments to verify the efficacy of BeNeDiff on a widefield calcium imaging dataset, where a head-fixed mouse performs a visual decision-making task across multiple sessions [Musall et al., 2018; 2019]. The neural subspace in BeNeDiff exhibits high levels of disentanglement and neural reconstruction quality, as evidenced by multiple quantitative metrics. By guiding the diffusion model to activate individual latent factors, we verify that the neural dynamics within the disentangled subspace provide interpretable and selective quantifications of the behaviors of interest (e.g., paw movements) across multiple brain regions. These results advance our understanding of neuro-behavioral relationships through the identification of fine-grained behavioral subspaces and the uncovering of disentangled neural dynamics.

To highlight our major contributions: (1) This is the first work to explore wide-field imaging across multiple brain regions of the dorsal cortex of head-fixed mice during a decision-making task using neural subspace analysis, rather than merely performing neuro-behavior decoding. We uncover disentangled neural representations for various behaviors. (2) To visualize the behavior dynamics within a disentangled neural subspace of each brain region, we develop a novel VDM-based interpretation tool that faithfully reflects behavior-related neural dynamics. It is essential to interpret the meaning of each neural latent dimension as well as the behavior dynamics it encodes.

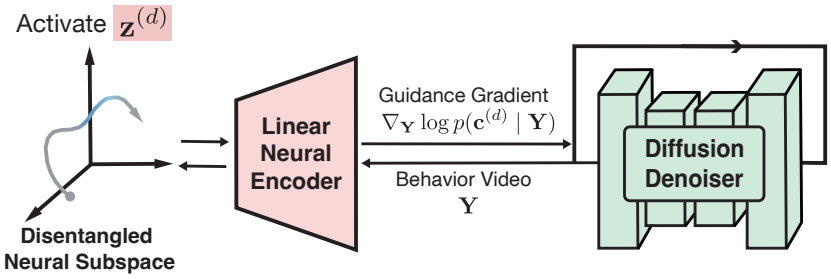

Figure 2: **Schematic diagram of neural dynamics interpretation with BeNeDiff**. We first employ a neural LVM to identify a disentangled neural latent subspace (the left part). Then, we train a linear neural encoder to map behavior video frames to neural trajectories. We use video diffusion models (VDMs) to generate behavior videos guided by the neural encoder, based on the objective of *activating the variance* of individual latent factors along the single-trial trajectory. This approach provides interpretable quantifications of neural dynamics in relation to the behaviors of interest.

2 Preliminaries

2.1 Problem Formulation

We first provide the notations of the paired neuro-behavioral observations. The single-trial neural population activities are denoted as $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_L]^\top \in \mathbb{R}^{L \times N}$, where L is the trial length (*i.e.*, number of time bins), N is the number of observed neural signals. The behavioral video frames are denoted as $\mathbf{Y} = [\mathbf{Y}_1, \ldots, \mathbf{Y}_L]^\top \in \mathbb{R}^{L \times H \times W}$, where H, W are the height and width of the compressed behavior video frames. We extract behavior labels $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_L]^\top \in \mathbb{R}^{L \times B}$ from the video frames using a behavior LVM [Whiteway et al., 2021]. B is the number of the behavior.

We build a variational autoencoder (VAE) [Kingma and Welling, 2013] to infer the neural latent trajectories $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_L]^\top \in \mathbb{R}^{L \times D}$, which are also informed by behavioral labels. D is the latent factor number. We denote its probabilistic encoder and decoder as $q_\psi(\mathbf{Z} \mid \mathbf{X}, \mathbf{U})$ and $p_\phi(\mathbf{X}, \mathbf{U} \mid \mathbf{Z})$, respectively. We denote the neural trajectory of a single latent factor as $\mathbf{z}^{(d)} = \mathbf{Z}_{:,d}$, where $d \in \{1, 2, \cdots, D\}$. Our primary goal is to investigate the neural dynamics of $\mathbf{z}^{(d)}$ through selectivity quantifications of its corresponding single-trial behavioral video data $\mathbf{Y}$.

2.2 Generative Video Diffusion Models

Diffusion models have also achieved impressive results in video synthesis over recent years [Ho et al., 2022b;a; Harvey et al., 2022]. VDMs process a fixed number of frames and factorize them over the temporal dimension via a deep neural network [Ho et al., 2022a; Harvey et al., 2022]. The training of VDMs starts from a forward process with a variance schedule $\{\beta_1, \ldots, \beta_T\}$, the noised sample $\mathbf{Y}_t$ follows the Gaussian conditional: $q(\mathbf{Y}_t \mid \mathbf{Y}_0) := \mathcal{N}(\mathbf{Y}_t; \sqrt{\bar{\alpha}_t}\mathbf{Y}_0, (1 - \bar{\alpha}_t)\mathbf{I})$, where $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. A denoising model $\hat{\boldsymbol{\epsilon}}_\theta(\cdot)$ is trained to reverse the forward process using a weighted mean squared error loss:

$$\mathcal{L}_{\text{VDM}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}, \mathbb{E}_{t \sim \mathcal{U}[0,T]} \left[w(\lambda_t) \left\| \boldsymbol{\epsilon} - \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}} (\mathbf{Y}_t, t) \right\|_2^2 \right], \tag{1}$$

in which time-steps t are uniformly sampled and $w(\lambda_t)$ is the weighting ratio. This loss function can be justified as optimizing a weighted variational lower bound on the data log-likelihood. In the sampling phase, we start from $\mathbf{Y}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{L \times H \times W})$ and perform step-by-step denoising,

$$\mathbf{Y}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{Y}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}} (\mathbf{Y}_t, t) \right) + \sigma_t \boldsymbol{\epsilon}_t, \tag{2}$$

where random noise perturbation $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{L \times H \times W})$ for timesteps $t > 1$, $\boldsymbol{\epsilon}_t = \mathbf{0}$ when $t = 1$, and $\sigma_t^2 = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t$.

3 Methods

Then, we train a linear neural encoder from the behavior video frames to the neural trajectories. We leverage video diffusion models (VDMs) to generate behavior videos guided by the neural encoder,

based on the objective of *activating the variance* of individual latent factors along the single-trial trajectory, providing interpretable quantifications of neural dynamics with respect to the behaviors of interest.

In this section, we first detail the process by which BeNeDiff infers a disentangled neural latent subspace. We then discuss the approach that BeNeDiff interprets the selectivity of neural dynamics of latent factors using the video diffusion model.

3.1 Behavior-Relevant and Disentangled Neural Latent Subspace Learning

Drawing inspiration from recent progress in the field of neural LVMs [Kingma et al., 2014; Klys et al., 2018], we employ a VAE to learn a disentangled neural subspace. The neural data $\mathbf{X}$ usually contains a good amount of information other than behavior [Hasnain et al., 2023], thus an unsupervised disentangled VAE won't effectively discover disentangled subspace with behavior only. Therefore, we introduce behavior labels $\mathbf{U}$ to inform the VAE to learn a latent subspace that better accounts for the variance related to behavior. We note that this technique is widely adopted in previous neuro-behavioral analysis works [Wang et al., 2024; Schneider et al., 2023; Gondur et al., 2023]. Notably, to enforce the disentanglement in the latent subspace, we incorporate a *total-correlation* (TC) penalty term [Chen et al., 2018] to enforce the VAE to find statistically independent latent factors in the semi-supervised setting. The VAE optimizes the following evidence lower bound (ELBO) [MacKay, 2003]:

$$\begin{aligned}\log p_{\boldsymbol{\phi}}(\mathbf{X},\mathbf{U}) \geq & \mathbb{E}_{q_{\boldsymbol{\psi}}(\mathbf{Z}|\mathbf{X},\mathbf{U})}\Big[\underbrace{\log p_{\boldsymbol{\phi}}(\mathbf{X}\mid\mathbf{Z})}_{\text{Neural Reconstruction}}+\underbrace{\log p_{\boldsymbol{\phi}}(\mathbf{U}\mid\mathbf{Z})}_{\text{Behavior Info.}}\Big]-\underbrace{\mathbb{D}_{\mathrm{KL}}\Big(q_{\boldsymbol{\psi}}(\mathbf{Z}\mid\mathbf{X},\mathbf{U})\Big\|p(\mathbf{Z})\Big)}_{\text{Regularization Term}} \\ & \underbrace{-\beta\,\mathbb{D}_{\mathrm{KL}}\Big(q_{\boldsymbol{\psi}}(\mathbf{Z}\mid\mathbf{X},\mathbf{U})\Big\|\prod_{d}q_{\boldsymbol{\psi}}(\mathbf{z}^{(d)}\mid\mathbf{X},\mathbf{U})\Big)}_{\text{Total Correlation}}=:-\mathcal{L}_{\mathrm{VAE}}(\boldsymbol{\phi},\boldsymbol{\psi})\end{aligned} \tag{3}$$

in which $\mathbf{z}^{(d)}$ denotes the neural trajectory of the d-th latent factor, and the value of β controls the strength of disentanglement penalty. However, the factorial density in this term is untractable in practice, so here we use the minibatch-weighted sampling estimator [Chen et al., 2018] to approximate the TC penalty term. We note that the variational autoencoder employs a sequential architecture [Fabius and Van Amersfoort, 2014] to capture the overall temporal dynamics along the single-trial trajectory $\{\mathbf{x}_l\}_{l=1}^{L}$, plugging bi-directional recurrent units [Schuster and Paliwal, 1997] into both the probabilistic encoder $q_{\boldsymbol{\psi}}(\cdot)$ and decoder $p_{\boldsymbol{\phi}}(\cdot)$.

3.2 Diffusion Guided Video Generation for Neural Dynamics Interpretation

3.2.1 Downside of Latent Manipulation for Interpreting Neural Dynamics

As for testifying the neural dynamics of a single disentangled latent factor $\mathbf{z}^{(d)}$ on the behavioral videos $\mathbf{Y}$, a straightforward attempt is to train a neural-net model to approximate the posterior distribution $p(\mathbf{Y}\mid\mathbf{Z})$ and then perform latent manipulation on each single latent factor. There are two major techniques to perform latent manipulation. The first is a naïve manipulation. This method manipulates a single subspace $\mathbf{z}^{(d)}$ while keeping the non-target latent factors fixed at arbitrary values. It then observes how the manipulation affects $\mathbf{Y}$. The induced changes in the videos reveal the dynamics encoded by $\mathbf{z}^{(d)}$. The second method uses classifier-free guidance [Ho and Salimans, 2022], where we allow the activated latent factor $\mathbf{z}^{(d)}$ to evolve while fixing non-target latent factors to arbitrary values. However, setting arbitrary values without knowing the true distributions of non-target subspaces can lead to unnatural distortions in generated videos, complicating the visualization and interpretation of genuine animal behavioral dynamics.

3.2.2 Behavioral Video Generation for Neural Dynamics Interpretation

So here we employ the video diffusion models (VDMs) to explore factor-wise neural dynamics through a generative manner, which is capable of maintaining temporal consistency and behavioral dynamics across frames. The primary goal is to perform behavior data generation conditioned on activating a single latent factor along the neural trajectory. Thus the resulting behavior video can

provide interpretable quantifications of the neural dynamics of factor $\mathbf{z}^{(d)}$. Specifically, we implement classifier guidance [Kawar et al., 2022]. By Bayes rule, we obtain the following posterior density and gradient [Mardani et al., 2023]:

$$p_{\boldsymbol{\theta},\boldsymbol{\lambda}}\left(\mathbf{Y}_t \mid \mathbf{c}\right) = p_{\boldsymbol{\theta}}(\mathbf{Y}_t) p_{\boldsymbol{\lambda}}\left(\mathbf{c} \mid \mathbf{Y}_t\right) / \, p(\mathbf{c}), \tag{4}$$

$$\nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\theta},\boldsymbol{\lambda}}\left(\mathbf{Y}_t \mid \mathbf{c}\right) = \underbrace{\nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\theta}}(\mathbf{Y}_t)}_{\text{Unconditional Gradient}} + \underbrace{\nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\lambda}}\left(\mathbf{c} \mid \mathbf{Y}_t\right)}_{\text{Guidance Gradient}}, \tag{5}$$

in which $\boldsymbol{\theta}, \boldsymbol{\lambda}$ are the parameter sets for the classifier and the denoising model, respectively. Note that t indicates the time step in the diffusion model. Our goal is to estimate the two terms on the RHS of Eq. (5) to perform conditional denoising in each step. We first approximate the density of the behavior video data through a standard denoising model $\hat{\boldsymbol{\epsilon}}_\theta\left(\mathbf{Y}_t, t\right)$ according to Eq. (1) since the first unconditional gradient term can be derived through it:

$$\nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\theta}}\left(\mathbf{Y}_t\right) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \hat{\boldsymbol{\epsilon}}_\theta\left(\mathbf{Y}_t, t\right). \tag{6}$$

For the calculation of the guidance term, we first train a linear neural encoder as the classifier from the behavior video data to the neural latent variables of the learned semi-supervised VAE subspace. We denote the estimated neural latent trajectories as $\hat{\mathbf{Z}}_t = \left[\hat{\mathbf{z}}_{t,1}, \ldots, \hat{\mathbf{z}}_{t,L}\right]^\top \in \mathbb{R}^{L \times D}$, in which:

$$\hat{\mathbf{z}}_{t,l} = \mathbf{W} \operatorname{vec}(\mathbf{Y}_{t,l}) + \mathbf{q}; \;\; \mathbf{q} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}\right), \tag{7}$$

where $1 \leq l \leq L$, $\hat{\mathbf{z}}_{t,l} \in \mathbb{R}^D$ denotes the estimated value of latent factors at time bin l and diffusion step t. The parameter set of the linear encoder $\boldsymbol{\lambda} = \{\mathbf{W}, \mathbf{Q}\}$. $\mathbf{W} \in \mathbb{R}^{D \times M}$ is the linear transformation matrix, $\mathbf{Q} \in \mathbb{R}^{D \times D}$ is the covariance matrix and $\operatorname{vec}(\cdot)$ represents vectorizing the two-dimensional video frame into column vector. After training the encoder, we fix all parameters and use it to construct the density $p_{\boldsymbol{\lambda}}\left(\mathbf{c} \mid \mathbf{Y}_t\right)$.

The class labels $\mathbf{c} \in \left\{\mathbf{c}^{(1)}, \mathbf{c}^{(2)}, \ldots, \mathbf{c}^{(D)}\right\}$, in which $\mathbf{c}^{(d)}$ is a one-hot column vector with a one at the d-th dimension and zeros elsewhere. Drawing inspiration from Noise-Contrastive Estimation [Gutmann and Hyvärinen, 2010], our guidance objective of the activation of latent factor d-th is formulated as maximizing the variance of the trajectory $\hat{\mathbf{z}}^{(d)}$ while minimizing the variance of the other latent factor trajectories in $\hat{\mathbf{Z}}$:

$$\log p_{\boldsymbol{\lambda}}\left(\mathbf{c}^{(d)} \middle| \mathbf{Y}_t\right) = \log \left[\frac{\exp\left(f_{\boldsymbol{\lambda}}^{+}\left(\hat{\mathbf{Z}}, \mathbf{c}^{(d)}\right)/\tau\right)}{\exp\left(f_{\boldsymbol{\lambda}}^{+}\left(\hat{\mathbf{Z}}, \mathbf{c}^{(d)}\right)/\tau\right) + \sum_{k=1}^{K} \exp\left(f_{\boldsymbol{\lambda}}^{-}\left(\hat{\mathbf{Z}}, \mathbf{c}^{(d)}\right)/\tau\right)} \right], \tag{8}$$

where $f_{\boldsymbol{\lambda}}^{+}\left(\hat{\mathbf{Z}}, \mathbf{c}^{(d)}\right) = \operatorname{Var}\left(\hat{\mathbf{Z}}\right) \mathbf{c}^{(d)}$ calculates the variance of the selected latent factor and $f_{\boldsymbol{\lambda}}^{-}\left(\hat{\mathbf{Z}}, \mathbf{c}^{(d)}\right) = \operatorname{Var}\left(\hat{\mathbf{Z}}\right) \mathbf{c}^{(j)}, \; j \sim \operatorname{Uniform}(\{1, 2, \ldots, D\} \backslash \{d\})$ calculates the variance of another sampled latent factor's trajectory. $\operatorname{Var}\left(\hat{\mathbf{Z}}\right) \in \mathbb{R}^{1 \times D}$ is a row vector where each element is the variance of every latent factor along the neural trajectory. τ is the temperature parameter. K is a hyperparameter controlling the number of sampled negative samples at each iteration.

The gradient $\nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\lambda}}\left(\mathbf{c}^{(d)} \mid \mathbf{Y}_t\right)$ is computed using automatic differentiation [Paszke et al., 2017]. Algorithm 1 describes the guided behavior video generation steps of our proposed framework BeNeDiff.

Algorithm 1: Generative Video Diffusion Model for Neural Dynamics Interpretation

Input: Condition label $\mathbf{c}^{(d)}$ for interpreting the neural dynamics of the d-th latent factor
Initiate $\mathbf{Y}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{L \times H \times W})$
for $t = T$ ***to*** 1 **do**
  $\hat{\boldsymbol{\epsilon}}'_{\boldsymbol{\theta},\boldsymbol{\lambda}}\left(\mathbf{Y}_t, t\right) = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\theta}}\left(\mathbf{Y}_t, t\right) - \sqrt{1-\bar{\alpha}_t} \nabla_{\mathbf{Y}_t} \log p_{\boldsymbol{\lambda}}\left(\mathbf{c}^{(d)} \mid \mathbf{Y}_t\right)$
  $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\boldsymbol{\epsilon}_t = \mathbf{0}$
  $\mathbf{Y}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{Y}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \hat{\boldsymbol{\epsilon}}'_{\boldsymbol{\theta},\boldsymbol{\lambda}}\left(\mathbf{Y}_t, t\right)\right) + \sigma_t \boldsymbol{\epsilon}_t$
end
Output: Generated behavior video $\mathbf{Y}_0$

4 Related Works

Disentangled Latent Subspace Learning. Neural LVMs is a fundamental framework which posits that single-trial neural population activities rely on low-dimensional "neural manifolds" [Gallego et al., 2018; Mitchell-Heggs et al., 2023; Li et al., 2023a; Hurwitz et al., 2021] and their extracted latent variables are successful in describing single-trial neural activities [Li et al., 2024c; 2022; 2024d; Liu et al., 2021; 2022; Li et al., 2024a]. Learning disentangled latent variables that uncover statistically independent latent factors [Chen et al., 2018] can provide enhanced robustness, interpretability, and controllability. Typically, this type of work involves adding auxiliary regularizer terms to enhance orthogonality [Mathieu et al., 2019] and reduce the total correlation [Chen et al., 2018] among the latent factors. In neuroscience, there have been studies focusing on the disentanglement of latent subspace within rich behavioral data [Whiteway et al., 2021; Shi et al., 2021]. However, our work is the first to discover interpretable and disentangled latent subspaces of wide-field imaging data.

Generative Diffusion Models. In recent years, diffusion models have achieved great success in generating high-quality images due to their expressivity and flexibility [Ho et al., 2020; Song et al., 2020a;b; Vahdat et al., 2021]. Moreover, for the more challenging task of video generation, there have been several explorations using diffusion models to address it. From a modeling perspective, the key concern is how to maintain temporal dynamics and consistency across frames. Most existing works [Ho et al., 2022b;a] extend the 2D U-Net architecture [Ronneberger et al., 2015; Song et al., 2024] to a 3D framework by considering the time axis. In this 3D framework, convolutions are performed in both spatial and temporal dimensions. Additionally, recent studies in neural computation have leveraged generative diffusion models to tackle domain-specific tasks, such as neural distribution alignment [Wang et al., 2024] and decoding visual stimulus from brain activities [Sun et al., 2024b;a]. Our work is the first to employ generative diffusion models for analyzing neuro-behavioral data relationships.

5 Experimental Results

5.1 Dataset Description

A head-fixed mouse performed a visual decision-making task while neural activity across the dorsal cortex was optically recorded using widefield calcium imaging [Musall et al., 2019; Churchland et al., 2019]. The mouse's behavior included both instructed and uninstructed movements. For behavioral data acquisition, two cameras captured video frames from both a side view and a bottom view. The dataset comprises 1126 trials conducted over two sessions, with 189 frames per trial at a frame rate of 30 Hz. Concurrently, neural activity was recorded at the same frame rate. The grayscale video frames were downsampled to 128×128 pixels. We extract 275 dimensions of neural signals from the high-dimensional widefield imaging data using the open-sourced LocaNMF decomposition toolkit [Saxena et al., 2020]. As shown in Figure 3, the behaviors of interest include the moving lick spouts, moving levers, the single visible right paw trajectories, and the movement of the jaw and chest, all tracked using DeepLabCut [Mathis et al., 2018].

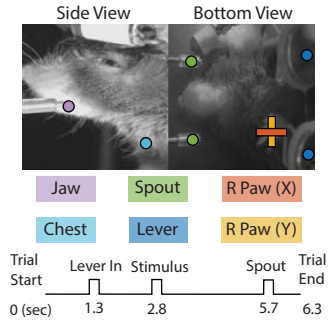

Figure 3: **Widefield Calcium Imaging Dataset**. The head-fixed mouse is performing a visual decision-making task, with the behaviors of interest and the trial structure illustrated.

5.2 Disentangled Neural Latent Subspace Investigation

We note that we train a unique neural LVM for each individual brain region (single-region), and we evaluate both the behavior decoding and neural reconstruction performance of each brain region-specific neural latent trajectories.

Single Latent Factor Behavior Decoding. In order to verify the disentanglement of the learned neural subspace in BeNeDiff, we evaluate the behavior label decoding performance of each individual latent factor. Specifically, we train a unique linear regressor for each latent factor from the VAE and plot the decoding accuracy as the R-squared value ($R^2\%$). The results of VIS-Right region (the right visual region) are shown in Figure 4. The main observation is that each latent factor is specific to a

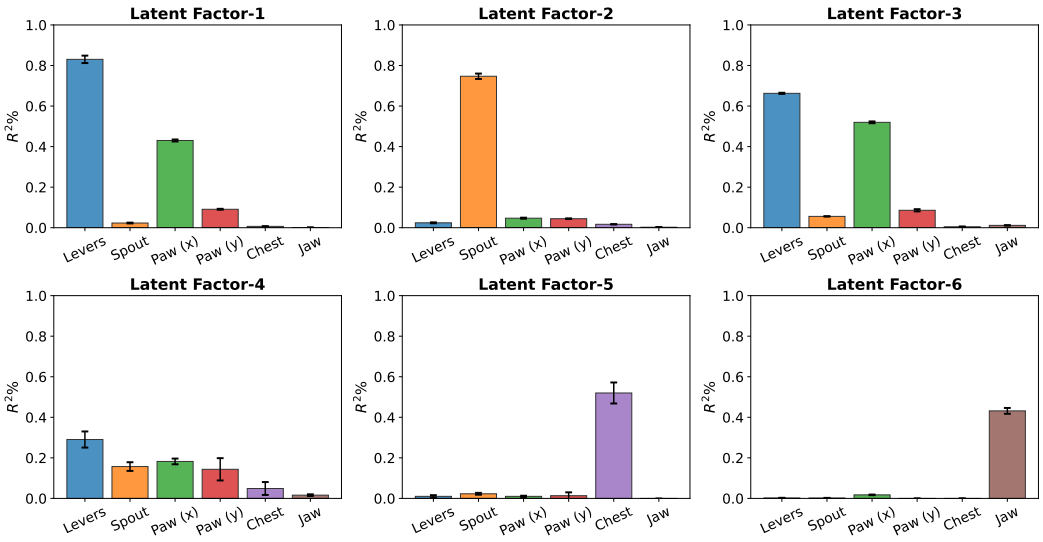

Figure 4: **Behavior decoding results of the disentangled neural latent variables of the VIS-Right region**. We observe that the decoding capability of each latent factor is specified to the corresponding behavior of interest, exhibiting a single-mode shape. In contrast, the original neural signals exhibit mixed selectivity to the behaviors, shown in Figure 1(B). Each experiment condition is repeated 5 times, with the mean represented by the bar plot and the standard deviations shown as error bars.

unique behavior of interest, confirming the orthogonality and clear disentanglement of the inferred latent trajectories from a quantitative perspective.

Neural Observation Signals Reconstruction. To prevent the VAE from overfitting to the behavioral labels, BeNeDiff also aims to maintain a low reconstruction error for neural activity. Table 1 presents the quantitative reconstruction results compared to baseline methods, including Semi-Supervised Learning (SSL) [Kingma et al., 2014], CEBRA [Schneider et al., 2023], and pi-VAE [Zhou and Wei, 2020]. The table records the R-squared values (R^2, in %) and RMSE for each method. Additionally, we plot the ground-truth neural signals and the reconstructed signals of several methods in a single trial in Figure 5. The main observation is that the neural reconstruction is well-preserved given the behavioral priors. One possible explanation is that the behavioral labels rotate the latent subspace while preserving the necessary information for reconstructing the neural data. The neural signals can be hardly recovered from the behavior labels only. It indicates that the behavior-informed latent does encode significant neural information that is not contained in the behavior labels. Furthermore, we evaluate the disentanglement quality of the latent subspace using the widely-adopted MIG (Mutual Information Gap) metric [Chen et al., 2018], also listed in Table 1. We observe that the learned latent subspace of BeNeDiff significantly enhances disentanglement compared to the vanilla VAE.

Table 1: **Baseline Comparison** of the neural LVM on two brain regions of Session-1. The boldface denotes the highest score of the MIG metric. Each experiment condition is repeated with 5 runs, and their mean and standard deviations are listed.

Region	Metrics	SSL	CEBRA	pi-VAE	**Ours**
VIS-Left	$R^2(\%)\uparrow$	81.10 ($\pm$0.26)	79.60 ($\pm$0.22)	74.37 ($\pm$0.24)	75.41 ($\pm$0.24)
	RMSE $\downarrow$	32.77 ($\pm$0.17)	33.07 ($\pm$0.18)	36.74 ($\pm$0.22)	35.50 ($\pm$0.17)
	MIG(%) $\uparrow$	37.50 ($\pm$0.20)	40.12 ($\pm$0.24)	43.98 ($\pm$0.29)	**55.87** ($\pm$0.26)
MOs-Left	$R^2(\%)\uparrow$	76.65 ($\pm$0.30)	72.63 ($\pm$0.28)	70.73 ($\pm$0.23)	69.59 ($\pm$0.22)
	RMSE $\downarrow$	30.64 ($\pm$0.21)	32.14 ($\pm$0.17)	35.69 ($\pm$0.19)	36.91 ($\pm$0.18)
	MIG(%) $\uparrow$	36.89 ($\pm$0.23)	37.94 ($\pm$0.23)	42.20 ($\pm$0.28)	**58.56** ($\pm$0.29)

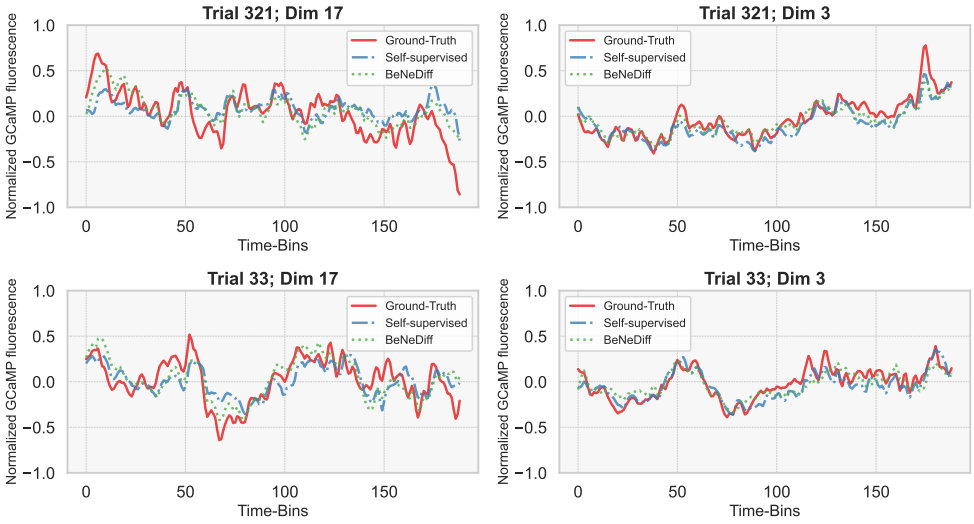

Figure 5: **Neural signal reconstruction performance evaluation of the VIS-Right region**. We observe that the neural reconstruction quality from the latent subspace of BeNeDiff is maintained given the behavioral labels. "Self-Supervised" denotes the VAE w/o behavior labels.

5.3 Neural Dynamics Exploration of Disentangled Latent Factors

From the quantitative experiments in the previous subsection, we obtained information about the decoding and disentanglement quality within the subspace. However, these metrics have limitations in interpreting single-trial neural dynamics, especially the complex temporal structures over time. Here, we visualize the generated videos from BeNeDiff and the baseline latent manipulation methods, demonstrating that BeNeDiff provides interpretable quantifications of the behaviors of interest.

Latent Manipulation Methods for Comparison. We compare the neural dynamics exploration performance of BeNeDiff against the following two latent manipulation methods:

- **Naïve Latent Manipulation**: the standard manipulation method discussed in Section 3.2.1, which approximates the posterior of behavioral videos given the neural latent trajectories $p(\mathbf{Y} \mid \mathbf{Z})$, using a neural network that incorporates recurrent units and spatio-temporal convolutional layers.
- **Classifier-free Guidance** [Ho and Salimans, 2022]: a method that approximates the posterior $p(\mathbf{Y} \mid \mathbf{Z})$ with diffusion models. It co-trains a conditional and an unconditional diffusion model together, combining the resulting conditional and unconditional scores at each diffusion step. In the conditional model, the entire neural latent trajectory $\mathbf{Z}$ is set as the condition, formulating the denoiser as $\hat{\boldsymbol{\epsilon}}\left(\mathbf{Y}_t, \mathbf{Z}, t\right)$. For the manipulation of the latent, we keep the activated latent factor $\mathbf{z}^{(d)}$ to evolve while setting the values of the other latent factors to those in the first frame of the trial.

Setup. To verify the neural dynamics interpretation capability of BeNeDiff, we generate behavioral video data given the activation of each behavior of interest (generated trials with the activation of Jaw and Paw-(y) are shown in Figure 6 and Figure 9, respectively). For visualization and video analysis, we plot frames at intervals of five and compute their frame differences. The conditional module of the classifier-free guidance method is trained with an auxiliary convolutional head. Compared to general video synthesis [Harvey et al., 2022; Esser et al., 2023], our behavioral video data are more focused on maintaining the temporal dynamics and consistency across video frames, thus in BeNeDiff, we tailor the standard 3D U-Net architecture [Çiçek et al., 2016] from temporal self-attention layers to temporal convolutions layers [Li et al., 2023b; 2024b] to maintain local temporal consistency. While we keep the spatial self-attention layers the same. The diffusion model is trained on an Nvidia V100, using approximately 20 computer hours.

Results Analyses of the Generated Videos. As shown in Figure 6, for the naïve latent manipulation method, the distribution of neural signals often falls outside the original distribution after manipulation, resulting in blurred generated frames. The frame differences are entangled, and the "Jaw" latent factor affects the entire head movement of the mouse, particularly in the first four frames shown.

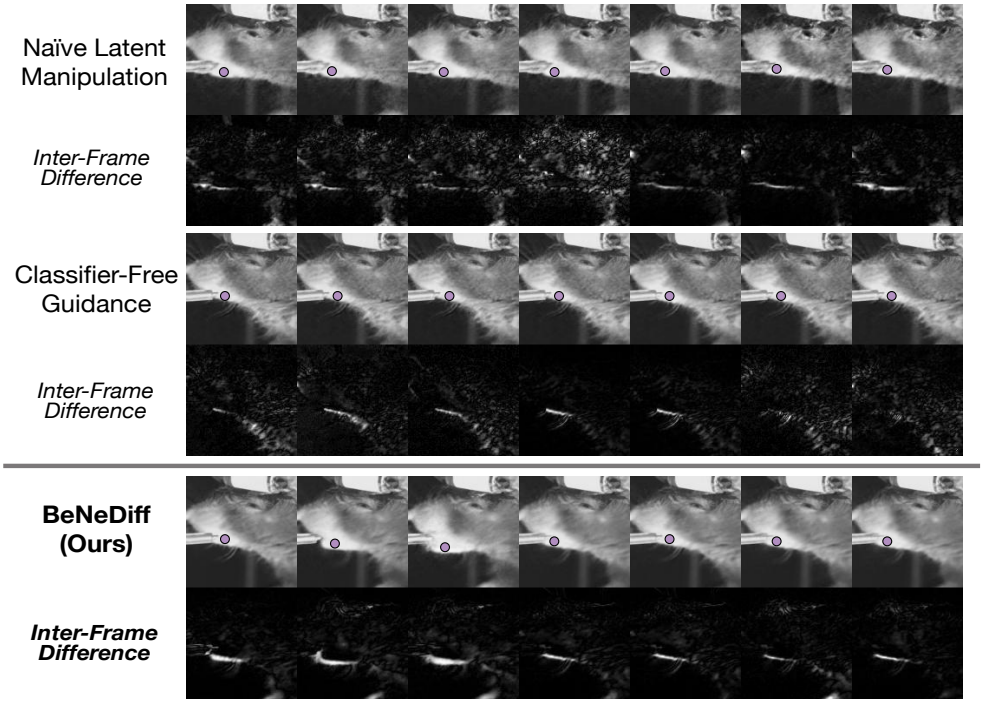

Figure 6: **Generated Single-trial Behavioral Videos with Latent Factor Guidance from the side view**. Compared to baseline methods, we observe that the neural dynamics of latent factor in the results of BeNeDiff show specificity to the "Jaw" movements.

On the other hand, for classifier-free guidance, the generated videos maintain coherent consistency between frames. However, it does not interpret neural dynamics well in this context, resulting in a trajectory with small movements in the "Jaw". This is because the overall latent trajectory is used as the input to the model and the other latent factors are kept fixed, making it difficult to discriminate the evolution of a single factor effectively. In contrast, the results of BeNeDiff show more specificity to the targeted behavior of interest. The inter-frame differences in BeNeDiff's results are clearly specified to the "Jaw" movements, and the structure of the neural dynamics is well-preserved and consistent with ground-truth "Jaw" behavior trajectories. A similar pattern is evident with the other latent factors, as shown in Figures 9, 10, and 11 in the appendix.

5.4 Neural Dynamics Exploration of Disentangled Latent Factors Across Brain Regions

Besides the capability of revealing interpretable neural dynamics of each latent factor associated with behaviors, here we further investigate the neural dynamics differences across brain regions through BeNeDiff. As shown in Figure 7 and Figure 12 in the appendix, we present the 2D neural latent trajectories of two latent factors, specifically related to "Paw-(x)" and "Paw-(y)", across six brain regions for two randomly selected trials. From the starting point of the trial, we observe that the latent trajectories corresponding to the left and right hemispheres of the VIS both show a noticeable change starting earlier. Next, the SSp regions show a large shift in activity, followed by a similar change in the MOs regions. However, it is difficult to clearly visualize the specific motion encoded by each region and to distinguish how different the motions are encoded solely based on neural trajectory plots. This further highlights the need for using a video diffusion model for visualization and interpretation.

In contrast, in the generated behavior video samples of BeNeDiff (as illustrated by the frame differences in Figure 8 and Figure 13 in the appendix), where the "Paw-(x)" and "Paw-(y)" latent factors are activated, the behavioral dynamics encoded by these two latent factors are observed across different brain regions. First, paw movements are detected in the VIS regions before the "Levers" come in. This early activity in VIS could reflect its role in the predictive coding of behaviors, indicating that this region may predict motor movements before they happen. Next, the SSp regions exhibit paw movements that are synchronized with the onset of the "Levers", indicating a potential role

for SSp in processing somatosensory feedback. Subsequently, in the MOs regions, paw movements are observed following the "Levers" onset, which is consistent with MOs' role in motor execution and control, occurring slightly after SSp.

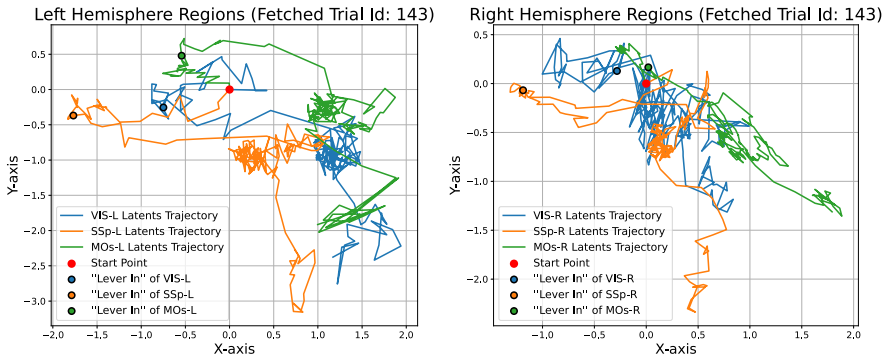

Figure 7: **Learnt Neural Latent Trajectories of BeNeDiff across various brain regions**. It is difficult to clearly visualize the specific motion encoded by each region and to distinguish how different the motions are encoded across brain regions.

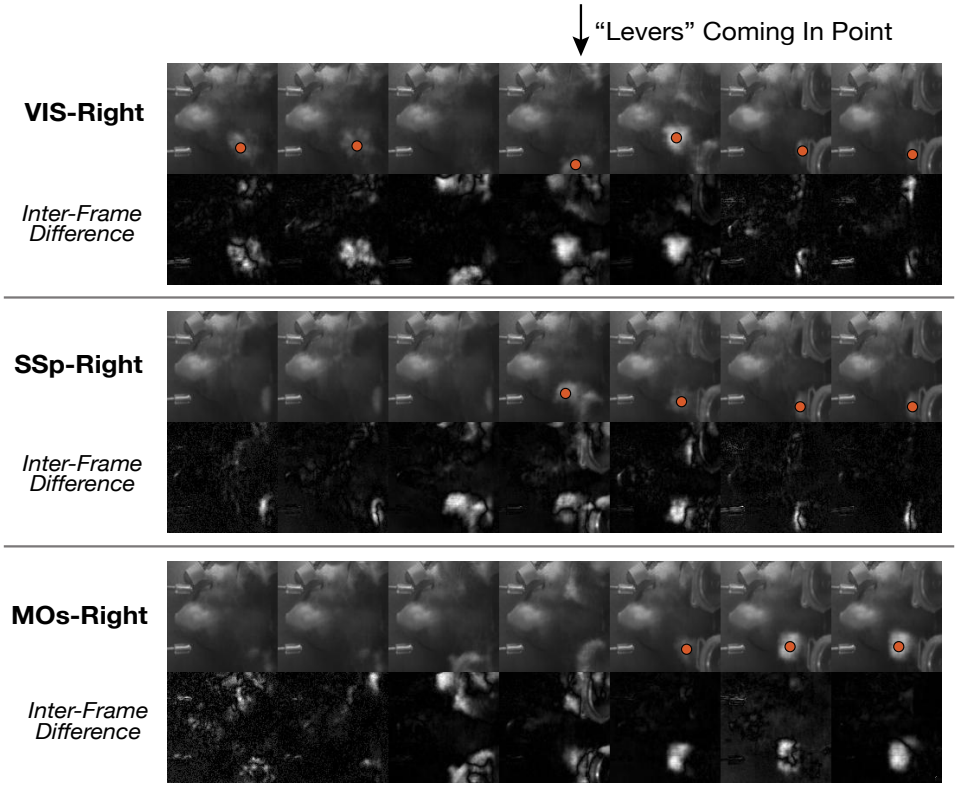

Figure 8: **Generated video frame differences across the right hemisphere regions**. The red dots in the figure indicate paw appearances.

To sum up, although neural trajectory plots provide a clear temporal sequence of activations across regions, it is challenging to directly visualize the specific behavioral dynamics encoded by each region and to discriminate how they differ. This limitation highlights the necessity for a video diffusion model in BeNeDiff, to better visualize and interpret the encoded behavioral dynamics of each neural latent factor. By synthesizing realistic behavior videos in a generative fashion, BeNeDiff enables us to better understand the unique neural dynamics in each brain region and their corresponding behavioral dynamics.

References

Eleanor Batty, Matthew Whiteway, Shreya Saxena, Dan Biderman, Taiga Abe, Simon Musall, Winthrop Gillis, Jeffrey Markowitz, Anne Churchland, John P Cunningham, et al. Behavenet: nonlinear embedding and bayesian neural decoding of behavioral videos. *Advances in Neural Information Processing Systems*, 32, 2019.

Gordon J Berman. Measuring behavior across scales. *BMC biology*, 16:1–11, 2018.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

Anne K Churchland, Simon Musall, Matthew T Kaufman, Ashley L Juavinett, and Steven Gluf. Dataset from: Single-trial neural dynamics are dominated by richly varied movements. 2019.

Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*, pages 424–432. Springer, 2016.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023.

Otto Fabius and Joost R Van Amersfoort. Variational recurrent auto-encoders. *arXiv preprint arXiv:1412.6581*, 2014.

Juan A Gallego, Matthew G Perich, Stephanie N Naufel, Christian Ethier, Sara A Solla, and Lee E Miller. Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature communications*, 9(1):4233, 2018.

Alex Gomez-Marin, Joseph J Paton, Adam R Kampff, Rui M Costa, and Zachary F Mainen. Big behavioral data: psychology, ethology and the foundations of neuroscience. *Nature neuroscience*, 17(11):1455–1462, 2014.

Rabia Gondur, Usama Bin Sikandar, Evan Schaffer, Mikio Christian Aoi, and Stephen L Keeley. Multi-modal gaussian process variational autoencoders for neural and behavioral data. *arXiv preprint arXiv:2310.03111*, 2023.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *Advances in Neural Information Processing Systems*, 35: 27953–27965, 2022.

Munib A Hasnain, Jaclyn E Birnbaum, Juan Luis Ugarte Nunez, Emma K Hartman, Chandramouli Chandrasekaran, and Michael N Economo. Separating cognitive and motor processes in the behaving mouse. *bioRxiv*, 2023.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022b.

Cole Hurwitz, Akash Srivastava, Kai Xu, Justin Jude, Matthew Perich, Lee Miller, and Matthias Hennig. Targeted neural dynamical modeling. *Advances in Neural Information Processing Systems*, 34:29379–29392, 2021.

Agustín Ibáñez, Adolfo M García, Sol Esteves, Adrián Yoris, Edinson Muñoz, Lucila Reynaldo, Marcos Luis Pietto, Federico Adolfi, and Facundo Manes. Social neuroscience: undoing the schism between neurology and psychiatry. *Social Neuroscience*, 13(1):1–39, 2018.

Bahjat Kawar, Roy Ganz, and Michael Elad. Enhancing diffusion-based image synthesis with robust classifier guidance. *arXiv preprint arXiv:2208.08664*, 2022.

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

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27, 2014.

Jack Klys, Jake Snell, and Richard Zemel. Learning latent subspaces in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

John W Krakauer, Asif A Ghazanfar, Alex Gomez-Marin, Malcolm A MacIver, and David Poeppel. Neuroscience needs behavior: correcting a reductionist bias. *Neuron*, 93(3):480–490, 2017.

Ed S Lein, Michael J Hawrylycz, Nancy Ao, Mikael Ayres, Amy Bensinger, Amy Bernard, Andrew F Boe, Mark S Boguski, Kevin S Brockway, Emi J Byrnes, et al. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445(7124):168–176, 2007.

Chengrui Li, Yule Wang, Weihan Li, and Anqi Wu. Forward \chiˆ2 divergence based variational importance sampling. *arXiv preprint arXiv:2311.02516*, 2023a.

Chengrui Li, Weihan Li, Yule Wang, and Anqi Wu. A differentiable partially observable generalized linear model with forward-backward message passing. In *Forty-first International Conference on Machine Learning*, 2024a.

Panfeng Li, Mohamed Abouelenien, and Rada Mihalcea. Deception detection from linguistic and physiological data streams using bimodal convolutional neural networks. *arXiv preprint arXiv:2311.10944*, 2023b.

Panfeng Li, Mohamed Abouelenien, Rada Mihalcea, Zhicheng Ding, Qikai Yang, and Yiming Zhou. Deception detection from linguistic and physiological data streams using bimodal convolutional neural networks. In *2024 5th International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*, pages 263–267. IEEE, 2024b. doi: 10.1109/ISPDS62779.2024.10667569.

Weihan Li, Yu Qi, and Gang Pan. Online neural sequence detection with hierarchical dirichlet point process. *Advances in Neural Information Processing Systems*, 35:6654–6665, 2022.

Weihan Li, Chengrui Li, Yule Wang, and Anqi Wu. Multi-region markovian gaussian process: An efficient method to discover directional communications across multiple brain regions. *arXiv preprint arXiv:2402.02686*, 2024c.

Weihan Li, Yule Wang, Chengrui Li, and Anqi Wu. Markovian gaussian process: A universal state-space representation for stationary temporal gaussian process. *arXiv preprint arXiv:2407.00397*, 2024d.

Ran Liu, Mehdi Azabou, Max Dabagia, Chi-Heng Lin, Mohammad Gheshlaghi Azar, Keith Hengen, Michal Valko, and Eva Dyer. Drop, swap, and generate: A self-supervised approach for generating neural activity. *Advances in neural information processing systems*, 34:10587–10599, 2021.

Ran Liu, Mehdi Azabou, Max Dabagia, Jingyun Xiao, and Eva Dyer. Seeing the forest and the tree: Building representations of both individual and collective dynamics with transformers. *Advances in neural information processing systems*, 35:2377–2391, 2022.

David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*, 2023.

Emile Mathieu, Tom Rainforth, Nana Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders. In *International conference on machine learning*, pages 4402–4412. PMLR, 2019.

Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature neuroscience*, 21(9):1281–1289, 2018.

Rufus Mitchell-Heggs, Seigfred Prado, Giuseppe P Gava, Mary Ann Go, and Simon R Schultz. Neural manifold analysis of brain circuit dynamics in health and disease. *Journal of Computational Neuroscience*, 51(1):1–21, 2023.

Simon Musall, Matthew T Kaufman, Steven Gluf, and Anne K Churchland. Movement-related activity dominates cortex during sensory-guided decision making. *BioRxiv*, page 308288, 2018.

Simon Musall, Matthew T Kaufman, Ashley L Juavinett, Steven Gluf, and Anne K Churchland. Single-trial neural dynamics are dominated by richly varied movements. *Nature neuroscience*, 22 (10):1677–1686, 2019.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Andrinandrasana David Rasamoelina, Fouzia Adjailia, and Peter Sinčák. A review of activation function for artificial neural network. In *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 281–286. IEEE, 2020.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

Omid G Sani, Hamidreza Abbaspourazad, Yan T Wong, Bijan Pesaran, and Maryam M Shanechi. Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. *Nature Neuroscience*, 24(1):140–149, 2021.

Shreya Saxena, Ian Kinsella, Simon Musall, Sharon H Kim, Jozsef Meszaros, David N Thibodeaux, Carla Kim, John Cunningham, Elizabeth MC Hillman, Anne Churchland, et al. Localized semi-nonnegative matrix factorization (locanmf) of widefield calcium imaging data. *PLoS computational biology*, 16(4):e1007791, 2020.

Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, 2023.

Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

Changhao Shi, Sivan Schwartz, Shahar Levy, Shay Achvat, Maisan Abboud, Amir Ghanayim, Jackie Schiller, and Gal Mishne. Learning disentangled behavior embeddings. *Advances in neural information processing systems*, 34:22562–22573, 2021.

Han Song, Cong Liu, and Huafeng Dai. Bundledslam: An accurate visual slam system using multiple cameras. In *2024 IEEE 7th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, volume 7, pages 106–111, 2024. doi: 10.1109/IAEAC59436.2024.10503743.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Charu Bai Reddy, Matteo Carandini, and Kenneth D Harris. Spontaneous behaviors drive multidimensional, brainwide activity. *Science*, 364(6437):eaav7893, 2019.

Jingyuan Sun, Mingxiao Li, Zijiao Chen, and Marie-Francine Moens. Neurocine: Decoding vivid video sequences from human brain activties. *arXiv preprint arXiv:2402.01590*, 2024a.

Jingyuan Sun, Mingxiao Li, Zijiao Chen, Yunhao Zhang, Shaonan Wang, and Marie-Francine Moens. Contrast, attend and diffuse to decode high-resolution images from brain activities. *Advances in Neural Information Processing Systems*, 36, 2024b.

Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in neural information processing systems*, 34:11287–11302, 2021.

Sandra Vieira, Walter HL Pinaya, and Andrea Mechelli. Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications. *Neuroscience & Biobehavioral Reviews*, 74:58–75, 2017.

Yule Wang, Zijing Wu, Chengrui Li, and Anqi Wu. Extraction and recovery of spatio-temporal structure in latent dynamics alignment with diffusion model. *Advances in Neural Information Processing Systems*, 36, 2024.

Matthew R Whiteway, Dan Biderman, Yoni Friedman, Mario Dipoppa, E Kelly Buchanan, Anqi Wu, John Zhou, Niccolò Bonacchi, Nathaniel J Miska, Jean-Paul Noel, et al. Partitioning variability in animal behavioral videos using semi-supervised variational autoencoders. *PLoS computational biology*, 17(9):e1009439, 2021.

Ding Zhou and Xue-Xin Wei. Learning identifiable and interpretable latent models of high-dimensional neural activity using pi-vae. *Advances in Neural Information Processing Systems*, 33: 7234–7247, 2020.

Appendix to "Exploring Behavior-Relevant and Disentangled Neural Dynamics with Generative Diffusion Models"

A Methodology details

Broader Impacts and Future Work. Our results highlight the method's ability to reveal fine-grained neuro-behavioral relationships, advancing our understanding of how neural dynamics encode behavior. These results demonstrate how BeNeDiff can elucidate interpretable quantifications of behaviors of interest, making it a promising machine learning tool for explainable neuroscience. Future work will explore extending this approach to more neural datasets and further refining the generative models for more theoretical interpretability and utility in neuroscience research.

Training Details of Neural LVM. The neural signal dimensions for the brain regions are as follows: MOs_L: 14 dimensions, MOs_R: 14 dimensions, VIS_L: 24 dimensions, VIS_R: 21 dimensions, SSp_L: 23 dimensions, and SSp_R: 22 dimensions. Both the probabilistic encoder and decoder of the neural LVM are based on an RNN architecture Fabius and Van Amersfoort [2014]. Mean squared error (MSE) is used for both the neural reconstruction and behavior decoding loss. We use the Adam Optimizer Kingma and Ba [2014] for optimization and the learning rate is set as 0.001. The batch size is uniformly set to 32. The latent subspace factor number is fixed at 6, which is the same as the number of behaviors of interest. We employ the dropout technique Srivastava et al. [2014] and the ReLU activation function Rasamoelina et al. [2020] between layers in our probabilistic encoder and decoder neural networks.

Training Details of Video Diffusion Models. We adopt the architecture of the VDM of 3D-UNet [Ho et al., 2022b] with the ϵ-parameterization. We use both spatial attention and spatial convolutions. The temporal convolutions are used to maintain consistency between frames. The embedding input size to the UNet architecture is set as 32 and the UNet has three downsampling and upsampling layers. The diffusion timestep is set as 200. The training batch size is set as 64, with a learning rate of 0.001. We use Group Normalization.

B In-depth Investigation on the neural LVM module across brain regions

Table 2: The $R^2\%$ and RMSE of the neural reconstruction, and the disentanglement MIG of the latent subspace on the VIS-Right region data. The boldface denotes the highest score of the MIG metric. Each experiment condition is repeated with 5 runs, and their mean and standard deviations are listed.

Metrics \ Method	Session-1		Session-2	
	Standard VAE	**Ours**	Standard VAE	**Ours**
$R^2(\%)\uparrow$	77.79 ($\pm$0.20)	73.74 ($\pm$0.24)	78.68 ($\pm$0.21)	71.13 ($\pm$0.29)
RMSE $\downarrow$	48.94 ($\pm$0.18)	55.17 ($\pm$0.19)	49.54 ($\pm$0.22)	54.27 ($\pm$0.19)
MIG$(\%)\uparrow$	34.61 ($\pm$0.30)	**56.36** ($\pm$0.29)	33.20 ($\pm$0.27)	**59.05** ($\pm$0.26)

Table 3: **Ablation Study** of the neural LVM module. The boldface denotes the highest score of the MIG metric. Each experiment condition is repeated with 5 runs, and their mean and standard deviations are listed.

Region	Metrics	Standard VAE	w/o Beha	w/o TC	**Ours**
VIS-Left	$R^2(\%)\uparrow$	83.66 ($\pm$0.21)	77.74 ($\pm$0.23)	79.82 ($\pm$0.29)	75.41 ($\pm$0.24)
	RMSE $\downarrow$	30.96 ($\pm$0.18)	34.86 ($\pm$0.20)	34.71 ($\pm$0.13)	35.50 ($\pm$0.17)
	MIG$(\%)\uparrow$	33.13 ($\pm$0.24)	48.54 ($\pm$0.23)	38.13 ($\pm$0.27)	**55.87** ($\pm$0.26)
MOs-Left	$R^2(\%)\uparrow$	84.70 ($\pm$0.24)	76.08 ($\pm$0.20)	75.49 ($\pm$0.22)	69.59 ($\pm$0.22)
	RMSE $\downarrow$	31.41 ($\pm$0.22)	34.14 ($\pm$0.25)	34.92 ($\pm$0.16)	36.91 ($\pm$0.18)
	MIG$(\%)\uparrow$	32.96 ($\pm$0.21)	49.79 ($\pm$0.23)	40.74 ($\pm$0.23)	**58.56** ($\pm$0.29)

C Video Generation Results on Various Behaviors of Interests

Using Figure 9 as an example, for the naïve latent manipulation method, the generated frames are in a reasonable form. Nevertheless, the frame differences are still intertwined, and the latent factor of "Paw-(y)" heavily affects the "Spout" movement. Meanwhile, for classifier-free guidance, the trajectories focus on the mouse movements, but they are still entangled with the "Chest" movements. In contrast, the results of BeNeDiff show more specificity to the targeted behavior of interest. The inter-frame differences in BeNeDiff's results are clearly specified to the "Paw-(y)" movements, and the temporal evolution of the neural dynamics is coherent with real-world mouse paw trajectories. The generated results in Figures 10 and 11 show a similar trend, demonstrating specificity to the Paw-(x)" and Spout" factors.

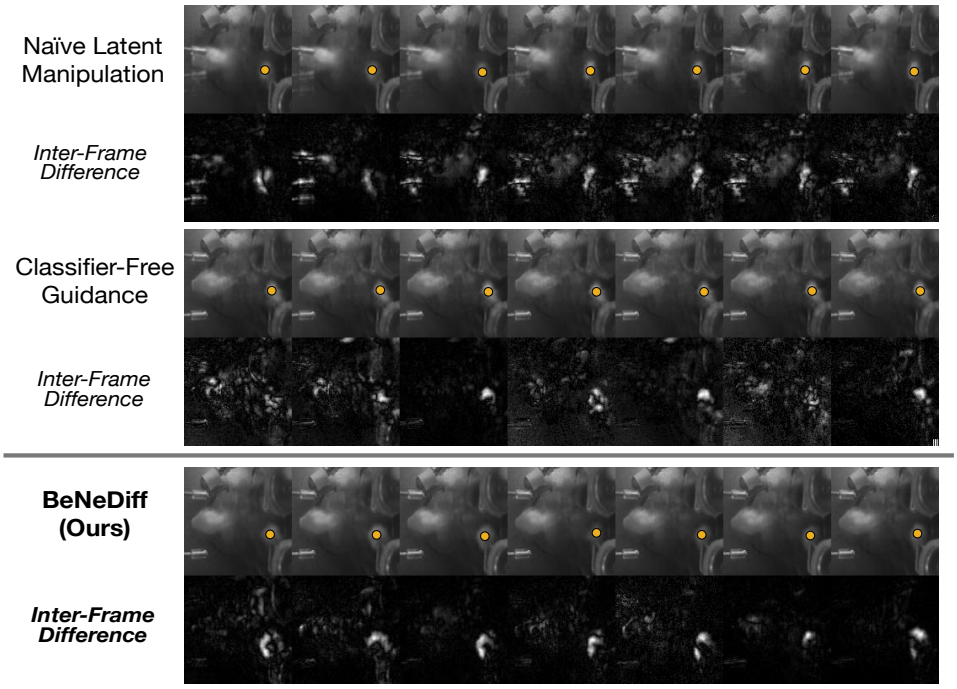

Figure 9: **Generated Single-trial Behavioral Videos with Latent Factor Guidance from the bottom view**. Compared to baseline methods, we observe that the neural dynamics of a latent factor in the results of BeNeDiff show specificity to the "Paw-(y)" movements.

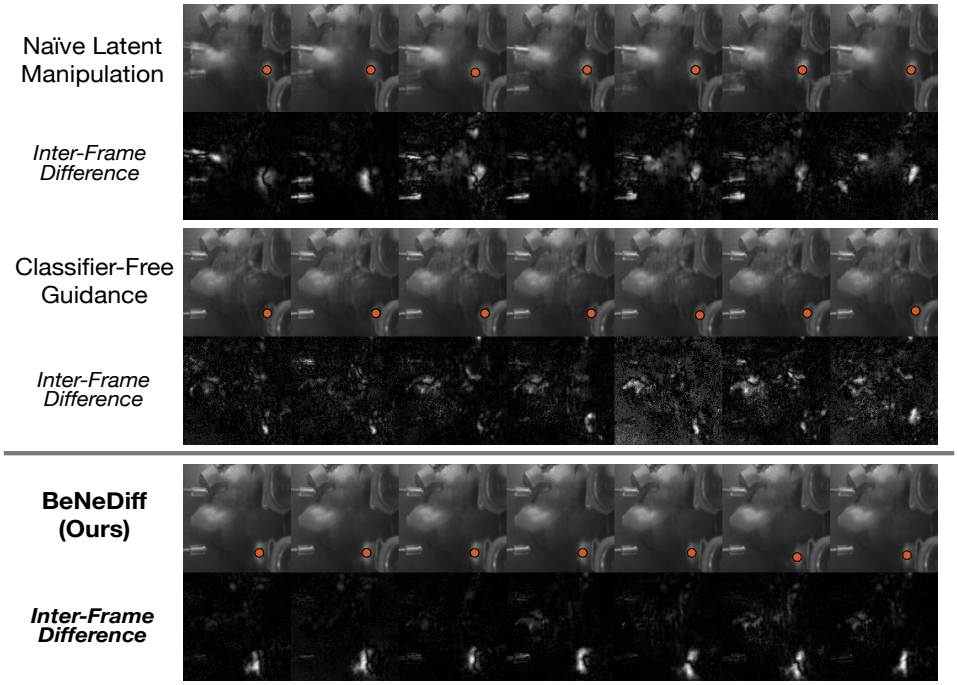

Figure 10: **Generated Single-trial Behavioral Videos with Latent Factor Guidance from the bottom view**. Compared to baseline methods, we observe that the neural dynamics of a latent factor in the results of BeNeDiff show specificity to the "Paw-(x)" movements.

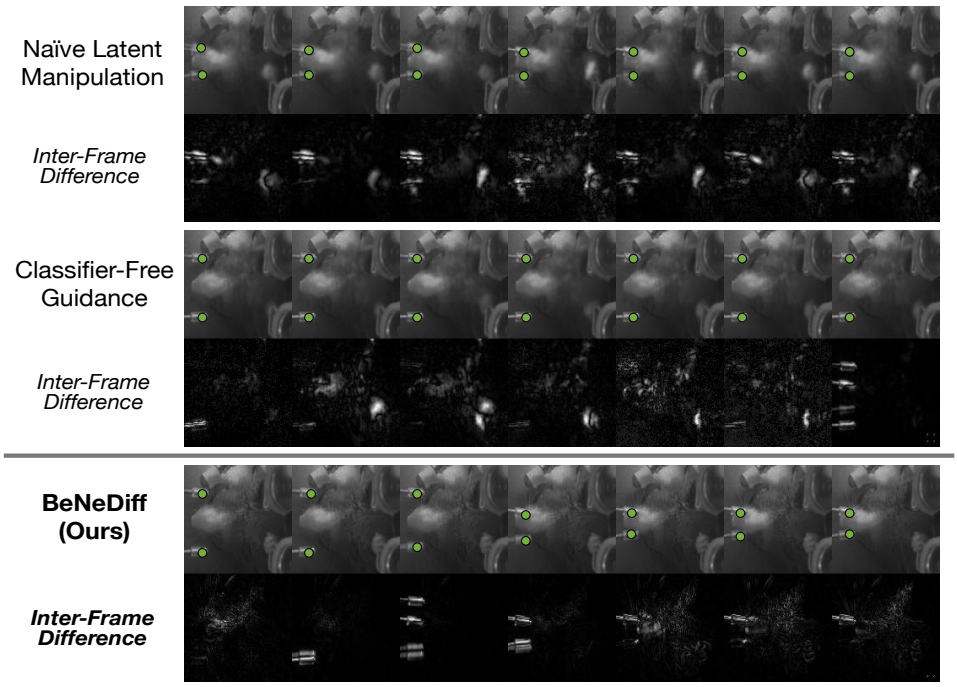

Figure 11: **Generated Single-trial Behavioral Videos with Latent Factor Guidance from the bottom view**. Compared to baseline methods, we observe that the neural dynamics of a latent factor in the results of BeNeDiff show specificity to the "Spout" movements.

D Learnt Neural Latent Trajectories of BeNeDiff across various brain regions

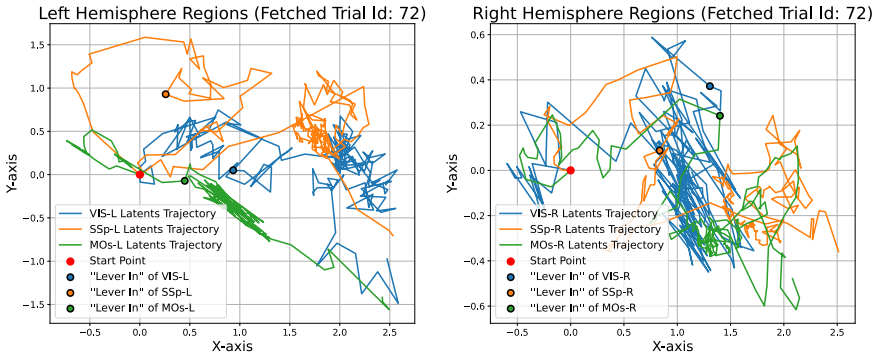

Figure 12: **Learnt Neural Latent Trajectories of BeNeDiff across various brain regions**. It is difficult to clearly visualize the specific motion encoded by each region and to distinguish how different the motions are encoded across brain regions.

E Video Generation Results on Various Brain Regions of the Left Hemisphere

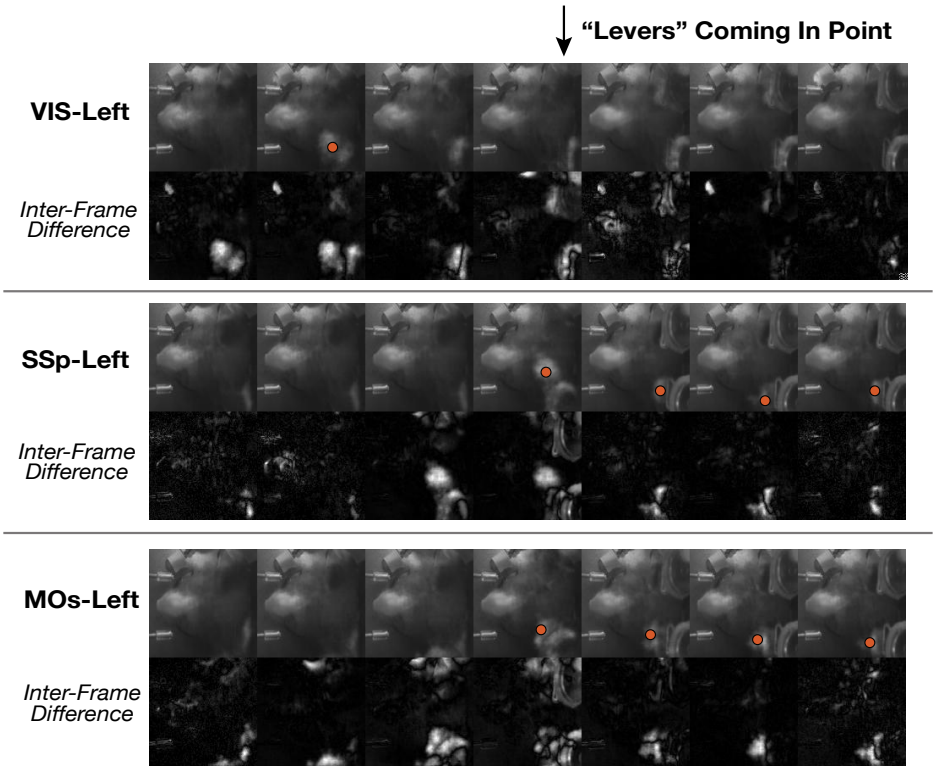

Figure 13: **Generated video frame differences across the left hemisphere regions**. The red dots in the figure indicate paw appearances.

F Discussion and Limitation

Our study introduces BeNeDiff, a novel approach leveraging behavior-informed latent variable models and generative diffusion models to uncover and interpret neural dynamics. Through empirical validation, we demonstrate that BeNeDiff effectively identifies a disentangled neural subspace and synthesizes behavior videos that provide interpretable insights into neural activities associated with distinct behaviors of interest. However, for the neural latent variable model (LVM) module, there exists a balance between disentangling the neural subspace with behavior semantics and maintaining neural reconstruction performance. For each brain region and session, at this stage, a careful hyper-parameter search is necessary to balance the weight between these two components. For the generative video diffusion module, we implement the neural encoder (classifier for guidance) as a linear regressor for interpretability. This linear assumption can be relaxed later for improved guidance performance.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In Section 1 Introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Section F Discussion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In Section 5 Experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: In Section 5 Experiments.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: In Section 5 Experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In Section 5 Experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In Section 5 Experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: [Yes]

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: In Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.