TEMPORAL DYNAMIC MODEL FOR RESTING STATE FMRI DATA: A NEURAL ORDINARY DIFFERENTIAL EQUATION APPROACH

A PREPRINT

Zheyu Wen
Department of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48105
zheyw@umich.edu

November 17, 2020

ABSTRACT

The objective of this paper is to provide a temporal dynamic model for resting state functional Magnetic Resonance Imaging (fMRI) trajectory to predict future brain images based on the given sequence. To this end, we came up with the model that takes advantage of representation learning and Neural Ordinary Differential Equation (Neural ODE) to compress the fMRI image data into latent representation and learn to predict the trajectory following differential equation. Latent space was analyzed by Gaussian Mixture Model. The learned fMRI trajectory embedding can be used to explain the variance of the trajectory and predict human traits for each subject. This method achieves average 0.5 spatial correlation for the whole predicted trajectory, and provide trained ODE parameter for further analysis.

1 Introduction

Figuring out spatial temporal relationship in functional Magnetic Resonance Imaging (fMRI) trajectory is a grand challenging problem. The field lacks a explainable and accurate model to fit the measured data. We are going to provide a new method to fit the fMRI trajectory. There are two main challenges in the field. First, we don’t know what’s the process of resting state fMRI data trajectory (Ligeois et al. [2017]). Second, it’s difficult to eliminate random noise and physiological noise in data, which makes the patterns more complex (Laumann et al. [2017]). AutoRegressive model (Zalesky et al. [2014]) and Hidden Markov Model (Vidaurre et al. [2017]), the two well known methods for explaining temporal dynamic, are far from understanding the process well.

Many work are dedicated to analyzing the spatial temporal dynamic regarding the measured fMRI data (Lurie et al. [2020], Chang and Glover [2010], Ligeois et al. [2017]). Some intrinsic properties are exploited from the observation data without an explicit mathematical model to interpret. No explainable and accurate model is proposed except some models based on Region Of Interest (ROI) with linear/non-linear Gaussian hypothesis (Ligeois et al. [2017]). Those works don’t provide a exploration of latent representation of measured fMRI data for spatial temporal information, which can be utilized to model the whole brain data. Neural network can help explore the principle of transition between spatial temporal representation with a predefined network architecture. Our work provide a interpretable model that can help predict future brain map or what happened between two given trajectories to interpolate for original data. The main idea is to compress the trajectory into spatial temporal latent representation and use a backbone of video prediction model to constrain the representation and help do forward prediction. In this work, we are not going to give physiological interpretation of the temporal relationship of measured fMRI data but to provide a new way of spatial temporal modeling on fMRI trajectory.

Predicting what fMRI data will be in the future can be viewed as a video prediction problem. Video prediction is challenging because of its uncertainty (Jayaraman et al. [2018]). Network that predict video frames recursively will accumulate blurry in images which makes the prediction unusable after several time points. To mitigate this problem and make the prediction with high quality, representation learning can be used to compress the spatial temporal information
in the bottleneck and temporal dynamic model can take advantage of these representation to do forward prediction. Our contribution is that we propose to use Neural Ordinary Differential Equation (Neural ODE) as video prediction backbone combined with a spatial temporal representation learning scheme to learn latent information of a group of fMRI images.

The paper is organized as follows, in section 2, we compare our model with other representation learning and video prediction model. We will also compare the difference between traditional temporal dynamic model for resting state fMRI data with ours. In section 3, formulation of Neural ODE will be introduced and we will explore the usage of learned spatial temporal representation. In section 4, we specify the experimental setting and result. In section 5, we discuss the key factors of success in our model and the potential usage of it in temporal relationship exploration for fMRI data.

2 Related work

Time varying functional connectivity is a popular terminology in describing temporal relationship of fMRI data discussed in Lurie et al. [2020] and Liegeois et al. [2017]. There are two popular methods in temporal dynamic modeling for resting state fMRI data which are Auto Regressive (AR) model and Hidden Markov Model (HMM). Liegeois et al. [2017] and Zalesky et al. [2014] used AR model as a simple linear model to fit the trajectory of ROI of resting state data and generate null data to test whether dynamic functional connectivity is significant enough. HMM is used to find the hidden state underneath the data. Vidaurre et al. [2017] found there are two meta-states among tens of state got from HMM. However these methods are all based on ROI which can not recover the whole brain activity after establishing the temporal model. Previous study also introduce PCA and ICA to compress the whole data into a 1D array and analyze based on this. Kim et al. [2020] introduced beta Variational AutoEncoder to compress the transformed 2D image into 1D array while disentangling the latent factors into different latent variables. Inspired from this, we develop a method to compute the spatial temporal representation to model the whole trajectory for each subject. Other methods used to explore functional connectivity dynamic are introduced in work of Cabral et al. [2017], Kashyap and Keilholz [2019], Chang and Glover [2010] and Laumann et al. [2017].

In our setting, we are going to predict future fMRI images based on the given sequence, which can be viewed as a video prediction problem. This problem has two main categories, which are forward prediction and bidirectional prediction. Making future prediction while maintaining high quality of image is hard. There are mainly two ways to deal with this problem. The first method in Oh et al. [2015] learn the prediction in the image space. While Ranzato et al. [2014] and Jayaraman and Grauman [2015] learned the temporal dynamic information of video frame in latent space. Recently time agnostic video prediction proposed by Jayaraman et al. [2018] lend new view point to the field. In our model, we train the model to learn spatial temporal latent representation for a batch of images while adapting Neural ODE to fit the learned representation.

Neural ODE was first introduced in Chen et al. [2018]. Later on more machine learning method related with ODE appear, including Augmented ODE introduced in Dupont et al. [2019], second order ODE in Yildiz et al. [2019], and Stochastic Differential Equation (SDE) in Jia and Benson [2019] provide more methods to model the temporal relationship of given data. Here we adapt Augmented ODE in our experiment that enlarge the representation space of Neural ODE, which is beneficial for us to improve the quality of prediction and makes it possible to predict beyond the space of input fMRI data trajectory to account for the variation of complex data. Another method in using Neural Network to fit fMRI data was proposed in Kashyap and Keilholz [2020] Khazaee et al. [2017] and Du et al. [2018] explored fMRI data temporal relationship which is useful for disease diagnosis or providing insight into fMRI data trajectory as shown in Tagliazucchi et al. [2012]. In our work, we use the learned spatial temporal latent representation to do the human traits prediction. This method can also be useful in individual classification and critical point analysis.

3 Spatial Temporal modeling on rsfMRI data

Our goal is to fit observation rsfMRI data into Neural ODE Network to make it possible to predict the future fMRI data given the input trajectory. We can model it as a video prediction problem. In video prediction, the goal is to predict the future frames given the first several input frames. There are two basic tasks. The first one is to do the forward prediction, and the target is the all future frames. The second task is the bidirectional prediction. We are given the first and the last several frames, and the target is to do the interpolation. In this section, we will talk about how to train the Ordinary Differential Equation using neural network to fit resting state fMRI data trajectory. We will also explain how the variance of the future prediction is accounted in latent representation of trajectory. The latent representation will contribute to some downstream analysis.
3.1 Basic definition

Human Connectome Project (HCP) provides resting state fMRI data records for different subjects. Each subject provides 1200 time points volumetric data with data size $91 \times 109 \times 91$. Each Volumetric data corresponding to a frame in video prediction context. We will adapt the dimension reduction method used in Kim et al. [2020] to reduce the Volumetric data dimension into 2D data with size $192 \times 192$. We denote the 2D fMRI data frame of subject $i$ as $X_{i,j}$. Our aim is to use the given first $j$ video frames $X_{i,0:j-1}$ to predict the future frames of subject $i$, which is $X_{i,j:J}$, where $J$ is maximum number of time point in record.

3.2 Neural ordinary differential equation

Video prediction is a challenging problem since the prediction quality may drop dramatically after several time points prediction. The uncertainty in video frames also contribute to the difficulty of predicting accurate future frames. Here we first try to solve the problem of prediction quality by using latent representation and Neural Ordinary Differential Equation introduced in Dupont et al. [2019]. Uncertainty will be explained by Variational AutoEncoder in next subsection.

Since resting state fMRI data changes very slowly during recording time, we can first do the downsampling of the original 1200 time points into 400 time points to reduce the prediction length while not influencing the utility of our result. Moreover, we do a innovative data processing to group the whole fMRI data trajectory into 4 groups for each individual, in which each group has 100 data frames. As shown in Fig. 1a we first use encoder to compress the 3D concatenated data (the first dimension is time while the last two dimensions are height and width of the 2D image frame) into a 1D array with length 64 to represent the spatial temporal information for fMRI sequence in 100 time points, which is denoted as $z_0$. The reason of selecting 100 as the group size is that the length of the data can successfully reveal the spatial temporal information for a certain subject. The encoder introduced here can be written as

$$z_0 = f_\theta(X_{i,0:99})$$  \hspace{1cm} (1)

We will not include subscripts for $z_t$ to avoid mass in our notation but $z_0$ is determined by input data for a certain subject $i$. $f_\theta$ is encoder and $\theta$ is parameter of encoder network. Then we apply Neural ODE on the latent representation to forward predict $z_t (t \in \mathbb{N}_+)$ . The differential equation are shown as following

$$\frac{dz_t}{dt} = W_2\Phi(W_1[z_t,0] + b_1) + b_2$$  \hspace{1cm} (2)

in which $W_1$, $W_2$, $b_1$, $b_2$ is linear network parameter and $\Phi$ is nonlinear function. Here we concatenate $0$ with $z_t$ to enlarge its representation space following method of Augmented Neural ODE (Dupont et al. [2019]). By solving the differential equation, we can obtain the latent representation in the future by following

$$\hat{z}_{t_1+\Delta t} = z_{t_1} + \int_{t_1}^{t_1+\Delta t} \frac{dz_t}{dt} \big|_{t=t_1} dt$$  \hspace{1cm} (3)

Lastly, we decode the predicted spatial temporal representation to recover the data in image space following

$$\hat{X}_{t_1,t_1+100:(t+1)*100-1} = g_\phi(z_t)$$  \hspace{1cm} (4)
where $g$ is the decoder and $\phi$ is parameter of it. Remind that we have four groups of data for each individual, the first group is used as input while the left three is regarded as output. The loss function is written as following

$$Loss = \sum_{i=1}^{b} \sum_{t=0}^{3} ||\hat{X}_{i,t+100:(t+1)\times100-1} - X_{i,t+100:(t+1)\times100-1}||_2$$

(5)

where $b$ is batchsize in the training and $X_{i,t+100:(t+1)\times100-1}$ is ground truth of the estimated fMRI data trajectory. We use Mean Squared Error as training loss between decoded images with ground truth images.

**Spatial correlation:** In testing time, we do forward prediction given the first group of images. Spatial correlation is used to evaluate the performance of trained network

$$R(X, \hat{X}) = \frac{C_{x,\hat{x}}}{\sqrt{C_{x,x}C_{\hat{x},\hat{x}}}}$$

(6)

where $C$ is covariance matrix. $x, \hat{x}$ are vectorized form of $X$ and $\hat{X}$. The high spatial correlation value means the similarity between the estimated images and ground truth images.

**Bidirectional prediction:** While the above description focus on forward prediction, it is easy to generalize to bidirectional prediction. In this situation, for each subject, we only know the first 200 data and last 100 data in training. We can assign latent representation to the trajectory it belongs to so the three predicted latent representation will be decoded to three known groups of images. Neural ODE can interpolate the left 100 time point for us while fitting the data on both end of trajectory. The loss function will be tweaked as

$$Loss = \sum_{i=1}^{b} \sum_{t=0}^{3} ||\hat{X}_{i,t+100:(t+1)\times100-1} - X_{i,t+100:(t+1)\times100-1}||_2$$

(7)

Therefore, we use the spatial temporal latent representation for different groups of frames. The network do forward predictions or bidirectional prediction of frames by first predicting on latent representation and then decode to image space.

### 3.3 Variance explanation introduced by VAE

To further explain the variance in spatial temporal latent representation, we introduce the Variational AutoEncoder (VAE) following [Kingma and Welling 2013] here to model the latent space as prior Gaussian distribution. In previous subsection, we use AutoEncoder(AE) to encode and decode the fMRI data latent representation. Here we make a slight change of this part. The VAE encoder encodes the group of images into a latent distribution. We then adapt reparametrization tricks in [Kingma and Welling 2013] to sample from this distribution and get latent representation. The distribution established here explains the variance of latent code for different subjects. The encoder part will be changed into

$$[\mu_0, logvar_0] = f_0(X_{i,0:99})$$

$$z_0 = \text{reparam}(\mu_0, logvar_0)$$

(8)

(9)

where $\mu_0$ and $logvar_0$ are mean and log variance of spatial temporal latent distribution, and $z_0$ has the same meaning as in previous subsection, which is spatial temporal representation of the given trajectory. Since we model the prior distribution of latent space as white Gaussian, here we use KL divergence to minimize the loss between approximate posterior distribution $p(z_0|X_{i,0:99})$ and given $\mathcal{N}(0,1)$.

$$Loss = \sum_{i=1}^{b} \sum_{t=0}^{3} ||\hat{X}_{i,t+100:(t+1)\times100-1} - X_{i,t+100:(t+1)\times100-1}||_2 + D_{KL}(q(z|X_{i,0:99}), p(z))$$

(10)

in which $p(z)$ is prior Gaussian.

### 3.4 Other temporal dynamic model

In the model mentioned above, we use ODE as backbone to predict spatial temporal latent representation $z_t$ given $z_0$. Here we will try a different backbone and a pure RNN model without spatial temporal representation learning. For model shown in Figure [1] we replace ODE by Recurrent Neural Network(RNN) as shown in Figure [10a] RNN take $z_{t-1}$ as input and output $z_t$ with other setting the same. We also want to test effectiveness of the latent representation $z_t$, so we tried a intuitive RNN architecture shown in Figure [10b]. We called it pure RNN model, in which the RNN take fMRI 2D image in one time point as input and output the image in the next time point. So in pure RNN we do one time point forward prediction rather than predicting a group of fMRI images.
3.5 Gaussian mixture model analyze latent space

We model the prior distribution of $z_0$ as Gaussian distribution, while in Neural ODE, $z_t$ will not necessarily follow the Gaussian distribution. It’s hard to use one Gaussian to explain the latent space. So here we introduce Gaussian Mixture Model (GMM) as shown in [Reynolds, 2009] to help explain and do clustering to see the common patterns among all the groups of images. We model the whole latent space as a Gaussian Mixture.

$$p(z_t) = \sum_{k=1}^{K} \pi_k \mathcal{N}(z_t | \mu_k, \Sigma_k)$$ (11)

where $t = 0, 1, 2, 3$, and $\mu_k, \Sigma_k$ are the mean and variance of each Gaussian distribution, and $\pi_k$ is weight for this Gaussian distribution. We will use this model to do clustering on latent space to see how many Gaussian distributions’ mixture can best explain the latent space. The process of GMM clustering is as follows. For a specific latent code $z_t$ of time point $t$. Suppose probability of data $z_t$ in cluster $k$ ($k = 1, ..., K$) is $\gamma(s_{tk}) = p(s_{tk} = 1 | z_t)$. The log probability of data $z_t$ is

$$\ln p(z_t) = \sum_{k=1}^{K} \gamma(s_{tk}) \ln p(z_t)$$ (12)

$$= \sum_{k=1}^{K} \gamma(s_{tk}) \ln \frac{p(z_t, s_{tk} = 1)}{p(s_{tk} = 1 \mid z_t)}$$ (13)

$$= \sum_{k=1}^{K} \gamma(s_{tk}) \ln p(z_t, s_{tk} = 1) - \sum_{k=1}^{K} \gamma(s_{tk}) \ln \gamma(s_{tk})$$ (14)

We maximize the log probability of $z_t$ to iteratively update variable $\gamma(s_{tk}), \pi_k, \mu_k$ and $\Sigma_k$. EM algorithm is used here to solve this cluster problem. For more information of EM algorithm in our setting, see Appendix B.

3.6 Gradient flow analysis on ODE

Ordinary Differential Equation reveals gradient magnitude and direction of $z_t$ when given current $z_t$ value. We care about how the system looks like before and after training. We plot the gradient flow of first two latent variable in $z_t$ for random initialized ODE model and there are several equilibrium in the whole system. On the sub-diagonal, the gradient flow run fast towards negative value for both first two latent variables in latent code. The initialized system may not be stable and as prediction goes, latent code may diverge fast.

![Figure 2: gradient flow for random initialize ODE.](image)

We can compute the equilibrium by setting Equation 2 to 0 and got a equilibrium set $\{z_t | \frac{dz_t}{dt} = 0\}$. Latent code around equilibrium will also be analyzed and corresponding temporal correlation for decoded estimation fMRI trajectory will be computed as Pearson product-moment correlation coefficients.

$$R(x, y) = \frac{\sum (I(x)_i - \bar{I}(x))(I(y)_i - \bar{I}(y))}{\sqrt{\sum (I(x)_i - \bar{I}(x))^2} \sqrt{\sum (I(y)_i - \bar{I}(y))^2}}$$ (15)

where $I(x)_i$ is pixel value at coordinate $x$ for time point $i$ while $\bar{I}(x)$ is mean of pixel value in $x$ among all the time points. Let’s assume $x$ is the seed we select and fix. $y$ is varied to allow us compute temporal correlation for all the pixel with the seed in image.
4 Experiment

We have proposed a new way of video prediction in resting state fMRI data. We focused our evaluation on quality of frame generation and explanation of spatial temporal latent code. Two other architecture were implemented to be compared with. The architectures are available in appendix A. There are mainly three downstream tasks. First we used GMM to explain the geometry of spatial temporal latent space. Second we tried to see how temporal correlation changes along the given gradient flow provided by trained Neural ODE. Thirdly, we concerned how to use latent code to predict human traits of subjects in HCP data.

4.1 data preparation and model architecture

The data was collected from HCP 3T resting state fMRI data. We applied detrend, band pass filter and standardize the signal, then extracted grey matter of brain and apply the transformer following [Kim et al., 2020] to convert Volumetric data into 2D fMRI data. For single volumetric fMRI data, two 2D images were obtained for left half sphere and right half sphere separately. Three hundreds subjects were used to train the network, fifty subjects were used in validation and one hundred fifty subjects were used for testing.

Our target is to extract spatial temporal latent code for fMRI data. We had a spatial encoder for both left and right half sphere 2D data to extract a latent representation for spatial information. We first added a dimension of channel for these 2D pre-processed data and input to Conv2D layer with output channel 32, kernel size 8, stride 2 and padding 0, then we concatenated data from left sphere and right sphere on the channel dimension. The concatenated data went through four Conv2D layers with output channel size 128, 128, 256, 256, each layer had kernel size 4, stride 2 and padding 0. Then we resized the output of last layer to 1D array and used one linear layer to transform into a vector of length 256. This is the spatial latent code for each data frame. We then concatenated spatial latent code of 100 consecutive frames into a matrix with size $100 \times 256$. We further used 2 Conv2D layers with output channel 2, 4 separately to encode this spatial temporal code matrix. Each layer has kernel size 3, stride 2 and padding 1. These two layers extract the spatial temporal information in this latent representation matrix and go through a linear layer to output $z_0$ with length 64. ODE network follows Equation 2 with tanh as nonlinear function. torchdiffeq package was used to do forward and backward propagation for training Neural ODE.

The decoder architecture is symmetric to the encoder. We first did linear transformation to spatial temporal latent code and feed into two Convolutional transpose 2D layer with output channel size 2, 1, and padding 0, 1 separately. The kernel size is 3 and stride maintains 2. Then feed the spatial latent information into four 2D Convolutional transpose layers with padding changed to 3. Lastly the output is chunked into two part in its channel dimension and feed into another 2D Convolutional transpose layers with padding 1. Other setting in these two layers are symmetric with encoder.

4.2 Training and spatial correlation

![Figure 3: Loss curve and spatial correlation in training for three different model architecture. ODE and RNN with AutoEncoder will take $10^4$ epochs to converge and attain 0.6 spatial correlation. Pure RNN converges very fast and will achieve 0.8 spatial correlation for training data in 500 epochs.](image)

The training was run on the GPU server Tesla K80. We selected Adam optimizer with learning rate $10^{-4}$, $\beta_1 = 0.9$ and $\beta_2 = 0.99$. We focused our evaluation on training loss, spatial correlation on 2D image data. For training strategy of this network. We first trained the network to learn the spatial latent representation, which can be regarded as AutoEncoder.
Then we trained the network to learn the spatial temporal latent code and ODE network. We focused our training on spatial temporal latent code part. It takes $10^4$ epochs to train and the loss curve is plotted on top of Figure 3a. On lower panel of the Figure 3a it depicts the spatial correlation during training. The final training correlation is close to 0.6. The training is successful and maintain the average spatial correlation in a high level. After training, we tested on 150 subjects to see the spatial correlation versus time point. We found the spatial correlation for each time point is significantly high enough, and correlation value is around 0.53, which means the spatial temporal latent representation is learned well and ODE can be used as backbone to establish the temporal relationship between different latent code.

4.3 Compare to other architecture

We tested our spatial temporal representation learning on other two architectures by replacing the Neural ODE as RNN. The aim is to test whether the latent code is learned successful for different temporal model. One of the model architecture is shown in Appendix A Figure 10a and training loss curve, spatial correlation at training time is shown in Figure 3b. We got similar result compared with Neural ODE which means the spatial temporal latent representation is the key in success of video prediction in resting state fMRI data. We also tested a intuitive temporal modeling by using pure RNN to establish temporal relationship between frames of different time points. In pure RNN following architecture in Hochreiter and Schmidhuber [1997], we didn’t use spatial temporal latent code but just compute spatial latent code and predict the future frame one by one. We didn’t do grouping for the trajectory and treat each frame as the output of the network. The model architecture is listed in Appendix A Figure 10b. The model converges in 500 epochs and spatial correlation in training is high around 0.8 as shown in Figure 3c. However in testing time, the spatial correlation is low and drop very fast versus time. The result is as expected, since neither RNN and ODE can predict very long time sequence with high quality. What we should do is to compress the data not only in spatial but also in temporal to reduce prediction time points in RNN or ODE and improve prediction quality.

4.4 Forward and bidirectional prediction

We had 150 subjects data for testing. The test consisted three parts: forward prediction, bidirectional prediction and VAE resampling. The qualitative evaluation of forward prediction can be seen in the Figure 5. We had 100 concatenated transformed 2D fMRI data as input. They were compressed into spatial temporal latent code and were propagated forward to predict three subsequent latent codes $z_t$ following the given $z_0$. We decoded these three subsequent codes into 300 2D fMRI data. Here we just show the 5 example input images and 5 example output images. More quantitative evaluation can be see in Figure 4. The x-axis is output time points and y-axis is spatial correlation. We deleted the first few and last few time point in the first two subfigure since they are lower than average abnormally, which can be left for further study. Figure 4a reveals our model performance when ODE is used as backbone for video prediction. Spatial correlation is high throughout the whole prediction trajectory. Figure 4b and Figure 4c reveal the performance of the trained network shown in Appendix A. The result reflects the success of training and latent code contain useful information for further usage.

![Example Image](image_url)

(a) ODE model spatial correlation for test data.
(b) RNN model with AutoEncoder spatial correlation for test data.
(c) Pure RNN model spatial correlation for test data.

Figure 4: Spatial Correlation for three different model. The first two model achieve high correlation versus time point, and the performance is stable. Pure RNN model prediction quality drop fast and go to zero after 50 time points prediction.

The second test is bidirectional prediction which is trained following Equation 7. In test time, we were given the first 100 2D fMRI data. The model was asked to predict the whole trajectory following Equation 2 and 3. Figure 6 plots the training input and prediction output in image space. The first row is first several starting fMRI data and second row is the last few fMRI data in the same trajectory. The output is interpolation of the fMRI data in the middle of ‘Start’ and ‘End’.
4.5 Three cluster center best explain GMM model

In section 3.3, we used Gaussian Mixture model to describe the distribution of spatial temporal latent code. To select a suitable number of cluster, we computed two metrics which are Silhouette score and Jensen Shannon score as criteria. Silhouette score is a method to validate consistency within clusters of data. The score ranges from −1 to +1, where a high value indicates that the high dimensional latent code in each cluster is well assigned, and the clustering configuration is appropriate. The Jensen Shannon score is a method of measuring the similarity between two probability distributions. It is also known as total divergence to the average. We had to select a cluster number that minimize the Jensen Shannon score. As shown in Figure 8b, when cluster number is 3, Silhouette score is high while Jensen Shannon score is low. In Figure 8a we plot the latent code distribution and cluster ellipsoid in the first two dimension of high dimensional latent code.

To further explore the property of three cluster centers, we first decoded the cluster center into three groups of images with each group 100 fMRI data frames. Then we computed the seed based temporal correlation and plot in Figure 8c. Seed location is plotted as a red point in the first subplot, and followed by three temporal correlation map in image space. The first subplot is 2D images representing surface of the brain map that we extract from HCP Volumetric data following [Kim et al., 2020].
Figure 7: We model the latent representation as prior Gaussian distribution. Different sample will result in image sequence with slight difference. Red circle point out the difference caused by VAE model.

(a) Spatial temporal latent representation distribution in first two dimensions. Ellipsoids depict mixture of three Gaussian distribution. High Silhouette score and low Jensen score is preferred.

(c) Three seed based temporal correlation for three cluster centers shown in (a). Seed point is shown as red point in first subfigure.

Figure 8: Gaussian Mixture Model is performed in latent space for 150 subjects. We select cluster number that Silhouette score is high while Jensen score is low. Each cluster center is decoded into image space corresponding to 100 image frames each. Temporal correlation is computed to reveal the common pattern.

4.6 Latent code change along gradient flow

In section 3.6, we talked about the analysis on trained Neural ODE parameter. After training, we can compute the equilibrium of ordinary differential equation. The equilibrium is also a latent code. We want to know how the change of latent code lead the change of temporal correlation for groups of images. The latent code was changed along the gradient flow by a predefined step size, and we obtained 4 different latent codes along this gradient flow starting from equilibrium. The latent code was decoded into groups of images. Seed based temporal correlation was calculated for these four groups of images separately and are depicted in Figure 9. The temporal correlation map shows the decreasing correlation of other area against the seed we selected. This may help understand how the brain pattern change on the direction of gradient flow provided by the trained ODE model.
4.7 Human traits analysis using latent space

The spatial temporal latent code can be further used to estimate human traits for each subject in HCP data. We selected six human traits including passive traits and active tasks as target for our prediction. The value of each trait can be obtained accompanying with the record of fMRI data. We added one linear layer with latent code as input and output the estimation of different human traits. Then we calculate the error of prediction for these 6 different human traits by Normalized Root Mean Squared Error (NRMSE). The result is summarized in the table 1.

| Name       | FearAffect | Sadness| LifeSatif | Friendship | Language Task Acc | Relational Task Acc |
|------------|------------|--------|----------|------------|-------------------|---------------------|
| NRMSE      | 0.016      | 0.024  | 0.014    | 0.012      | 0.008             | 0.013               |

Table 1: Human traits prediction error

Our latent code can provide a very accurate prediction of the human traits for different subjects which means characteristic of fMRI trajectory for different subjects can be successfully revealed by latent code. There are very deterministic relationship between the trajectory information and human behavior.

5 Discussion

In this paper, we proposed a new method of video frame prediction for resting state fMRI data and made use of latent code to do some downstream tasks. The key point for success is the compression on both spatial and temporal information. The difference between Neural ODE and RNN is that in training time, Neural ODE is a recursive forward propagation method and output of first ODE block will be the input of the next ODE block. RNN is always used to do a one step forward prediction in training while in testing, it generate prediction recursively. Either ODE or RNN will work well regarding our spatial temporal latent code and differ little in performance. However ODE can be easily used to interpret how the latent code in one time stamp transferred to the next by analyzing differential equation while RNN lack this interpretability.

Besides the prediction quality, the geometry explanation of latent space and utilization of ODE trained parameters are two main concern in this paper. [Kim et al., 2020] explained geometry of spatial representation of resting state fMRI as a sphere in high dimension which differs for different subjects. While in forward prediction in our temporal dynamic model, the latent space would not necessarily be Gaussian distribution and would be influenced by both starting point of forward prediction and ODE trained parameter. In experiment, we analyzed the distribution of latent code of ODE prediction. The visualization of first two variables in latent code reveals that it is not Gaussian but can be explained by Gaussian mixture. According to Silhouette score and Jensen Shannon score, it revealed that three Gaussian Mixture can explain well to the resulting latent code distribution. The latent code was transferred from zero mean Gaussian to another Gaussian with different mean value, but the variance change little, which is due to the combination of prior Gaussian assumption for $z_0$ and ODE propagation for $z_t$.

We tested the trained model for forward prediction, bidirectional prediction and sampling from VAE latent space to get qualitative evaluation. The prediction looks good and VAE also explain the variance of the latent code and show in image space as shown in Figure 7. The region within the red circle may reveal the subject difference among whole population.

Another potential usage of this model in fMRI data is to do interpolation for smaller group of fMRI images. The measurement of fMRI data always include some random noise and physiological noise. [Power et al., 2014] proposed...
censoring method to eliminate the head motion in fMRI data that is significant. Laumann et al. [2017] used censoring method to help understand time varying temporal correlation but that method is criticized for it destroys the temporal relationship. But ODE can interpolate the censored data if we know which time points are censored. In work of Khazaee et al. [2017] and Du et al. [2018], the author used fMRI data to do diagnosis. Our spatial temporal latent code can also be used in disease diagnosis and human traits prediction as shown in section 4.7.

Overall, we treat the temporal dynamic modeling for fMRI data as a video prediction problem. The model accurately predicts the image frames given a group of images. Analysis on latent space and image space may shed light on the study of temporal dynamic of resting state fMRI data.

References

Joana Cabral, Morten L Kringelbach, and Gustavo Deco. Functional connectivity dynamically evolves on multiple time-scales over a static structural connectome: Models and mechanisms. *NeuroImage*, 160:84–96, 2017.

Catie Chang and Gary H Glover. Time–frequency dynamics of resting-state brain connectivity measured with fmri. *Neuroimage*, 50(1):81–98, 2010.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *Advances in neural information processing systems*, pages 6571–6583, 2018.

Yuhui Du, Zening Fu, and Vince D Calhoun. Classification and prediction of brain disorders using functional connectivity: promising but challenging. *Frontiers in neuroscience*, 12:525, 2018.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. In *Advances in Neural Information Processing Systems*, pages 3140–3150, 2019.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1413–1421, 2015.

Dinesh Jayaraman, Frederik Ebert, Alexei A Efros, and Sergey Levine. Time-agnostic prediction: Predicting predictable video frames. *arXiv preprint arXiv:1808.07784*, 2018.

Junteng Jia and Austin R Benson. Neural jump stochastic differential equations. In *Advances in Neural Information Processing Systems*, pages 9847–9858, 2019.

Amrit Kashyap and Shella Keilholz. Dynamic properties of simulated brain network models and empirical resting-state data. *Network Neuroscience*, 3(2):405–426, 2019.

Amrit Kashyap and Shella Keilholz. Brain network constraints and recurrent neural networks reproduce unique trajectories and state transitions seen over the span of minutes in resting-state fmri. *Network Neuroscience*, 4(2):448–466, 2020.

Ali Khazaee, Ata Ebrahimpazadeh, Abbas Babajani-Feremi, Alzheimer’s Disease Neuroimaging Initiative, et al. Classification of patients with mci and ad from healthy controls using directed graph measures of resting-state fmri. *Behavioural brain research*, 322:339–350, 2017.

Jung-Hoon Kim, Yizhen Zhang, Kuan Han, Minkyu Choi, and Zhongming Liu. Representation learning of resting state fmri with variational autoencoder. *bioRxiv*, 2020.

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

Timothy O Laumann, Abraham Z Snyder, Anish Mitra, Evan M Gordon, Caterina Gratton, Babatunde Adeyemo, Adrian W Gilmore, Steven M Nelson, Jeff J Berg, Deanna J Greene, et al. On the stability of bold fmri correlations. *Cerebral cortex*, 27(10):4719–4732, 2017.

Raphael Liegeois, Timothy O Laumann, Abraham Z Snyder, Juan Zhou, and BT Thomas Yeo. Interpreting temporal fluctuations in resting-state functional connectivity mri. *Neuroimage*, 163:437–455, 2017.

Daniel J Lurie, Daniel Kessler, Danielle S Bassett, Richard F Betzel, Michael Breakspear, Shella Kheilholz, Aaron Kucyi, Raphaël Liégeois, Martin A Lindquist, Anthony Randal McIntosh, et al. Questions and controversies in the study of time-varying functional connectivity in resting fmri. *Network Neuroscience*, 4(1):30–69, 2020.

Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. In *Advances in neural information processing systems*, pages 2863–2871, 2015.

Jonathan D Power, Anish Mitra, Timothy O Laumann, Abraham Z Snyder, Bradley L Schlaggar, and Steven E Petersen. Methods to detect, characterize, and remove motion artifact in resting state fmri. *Neuroimage*, 84:320–341, 2014.
Marc Aurelio Ranzato, Arthur Szlam, Joan Bruna, Michael Mathieu, Ronan Collobert, and Sumit Chopra. Video (language) modeling: a baseline for generative models of natural videos. arXiv preprint arXiv:1412.6604, 2014.

Douglas A Reynolds. Gaussian mixture models. Encyclopedia of biometrics, 741, 2009.

Enzo Tagliazucchi, Pablo Balenzuela, Daniel Fraiman, and Dante R Chialvo. Criticality in large-scale brain fmri dynamics unveiled by a novel point process analysis. Frontiers in physiology, 3:15, 2012.

Diego Vidaurre, Stephen M Smith, and Mark W Woolrich. Brain network dynamics are hierarchically organized in time. Proceedings of the National Academy of Sciences, 114(48):12827–12832, 2017.

Cagatay Yildiz, Markus Heinonen, and Harri Lahdesmaki. Ode2vae: Deep generative second order odes with bayesian neural networks. In Advances in Neural Information Processing Systems, pages 13412–13421, 2019.

Andrew Zalesky, Alex Fornito, Luca Cocchi, Leonardo L Gollo, and Michael Breakspear. Time-resolved resting-state brain networks. Proceedings of the National Academy of Sciences, 111(28):10341–10346, 2014.
Appendix A  Network architecture of RNN w/ and w/o AE

(a) ODE is replaced by RNN to predict the spatial temporal latent representation. Output of first RNN will be the input of the next RNN. Latent code is decoded to recover the data in image space.

(b) Traditional RNN architecture to predict spatial latent code. The output of RNN represents only one time points rather than groups of images. In test time, the prediction is recursively generated.

Figure 10: We tried two other architecture for forward prediction of fMRI data. The key success of high quality prediction is do compression on data not only in spatial space but also in temporal space. The architecture on the right can predict one step forward very accurate but performance drop quickly.

Appendix B  EM algorithm in GMM

We initialized means \( \mu_k \), covariances \( \Sigma_k \), and mixing coefficients \( \pi_k \) for K Gaussians in Equation[12]

E step:

\[
\gamma(s_{tk}) = \frac{\pi_k \mathcal{N}(\vec{x}_t | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\vec{x}_t | \mu_j, \Sigma_j)}
\] (16)

M step:

\[
\pi_k^{new} = \frac{N_k}{N} = \frac{\sum_{n=1}^{N} \gamma(s_{tn})}{N}
\] (17)

\[
\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(s_{tn}) z_t
\] (18)

\[
\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(s_{tn})(z_t - \mu_k^{new})(z_t - \mu_k^{new})^T
\] (19)

After several steps of EM, we could compute the mean and variance for each cluster. We decoded the clustering center to analyze the common brain patterns embedded in spatial temporal latent code.