

Neural Decoding and Understanding via Deep Learning

THESIS

Submitted in Partial Fulfillment of  
the Requirements for  
the Degree of

MASTER OF SCIENCE (Computer Engineering)  
at the

NEW YORK UNIVERSITY  
TANDON SCHOOL OF ENGINEERING

by

Chenqian Le

May 2024

# Neural Decoding and Understanding via Deep Learning

THESIS

Submitted in Partial Fulfillment of  
the Requirements for  
the Degree of

MASTER OF SCIENCE (Computer Engineering)

at the

NEW YORK UNIVERSITY  
TANDON SCHOOL OF ENGINEERING

by

Chenqian Le

May 2024

Approved:



---

Advisor Signature  
May 10, 2024

Date



---

Department Chair Signature  
May 10, 2024

Date

University ID: N14369631

Net ID: cl6707

Approved by the Guidance Committee:

Major: Computer Engineering




---

**Yao Wang**  
Professor  
NYU Tandon School of Engineering

**May 10, 2024**

---

Date



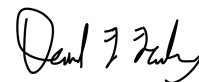
---

**Adeen Flinker**  
Associate Professor  
NYU Grossman School of Medicine

**May 12, 2024**

---

Date



---

**David Fouhey**  
Assistant Professor  
NYU Tandon School of Engineering

**May 14, 2024**

---

Date

Microfilm or other copies of this dissertation are obtainable from

UMI Dissertation Publishing

ProQuest CSA

789 E. Eisenhower Parkway

P.O. Box 1346

Ann Arbor, MI 48106-1346

## Vita

Chenqian Le was born in Jiangxi, China in 2000. He received his Bachelor's degree in Built Environment and Energy Engineering from Southeast University, Nanjing, China in 2022. He then continued his education at New York University Tandon School of Engineering starting in September 2022. He studied this thesis in NYU Video Lab from Jan 2023 to May 2024.

## Acknowledgements

I am deeply grateful to my advisor, Prof. Yao Wang, for her invaluable support and guidance since I joined the Video Lab. I also extend my heartfelt thanks to Prof. Adeen Flinker for his generous guidance and advice throughout my research journey. My appreciation further goes to Prof. David Fouhey for his insightful contributions and advice as a member of my committee.

I am thankful to Dr. Ran Wang and Dr. Junbo Chen for their encouragement and advice, which have been instrumental both in my research and in life. I also want to thank Dr. Amirhossein Khalilian-Gourtani for his insightful suggestions on my work. Special thanks are due to Xupeng Chen for his unwavering help and support, which have been invaluable to me. I appreciate the support and guidance from Chris Liu, and the collaborative efforts and advice from Nika Emami, which have enriched my research experience. I also appreciate advice from Antoine Ratouchniak.

Lastly, I wish to express my profound gratitude to my parents and my girlfriend, Qi You, for their unwavering support and encouragement.

May 2024

**ABSTRACT****Neural Decoding and Understanding via Deep Learning**

by

**Chenqian Le****Advisor: Prof. Yao Wang****Submitted in Partial Fulfillment of the Requirements for  
the Degree of Master of Science (Computer Engineering)****May 2024**

This thesis introduces innovative deep learning techniques for neural decoding and understanding, applying these methods across auditory and visual modalities. The research begins by examining the disentanglement of neural representations of speech through enhanced Swap Autoencoder architectures that incorporate a mix of data augmentation and hybrid neural networks. It then extends to real-time neural speech decoding with voice conversion, aimed at improving the accuracy of speech synthesis from brain activity. Additionally, the thesis explores the application of pre-trained deep neural networks to decode brain activity in

response to visual stimuli, which significantly enhances the robustness and accuracy of neural decoding systems. These studies present preliminary results that suggest promising directions for further research. The ongoing investigation will focus on refining these methodologies and exploring their implications for the development of more effective and interpretable brain-computer interfaces.



# Contents

Vita . . . . .	iv
Acknowledgements . . . . .	v
Abstract . . . . .	vi
List of Figures . . . . .	xiii
List of Tables . . . . .	xiv
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Related Work . . . . .	2
1.2.1 SwapVAE . . . . .	2
1.2.2 Speech Resynthesis . . . . .	3
1.2.3 ECoG to Text Decoding . . . . .	4
1.2.4 fMRI Decoding Model . . . . .	4
<b>2 Disentangled Neural Speech Representational Learning</b>	<b>6</b>
2.1 Introduction . . . . .	6
2.2 Encoder Backbone . . . . .	7
2.3 Swap Mechanism . . . . .	9
2.4 Unsupervised Swap Autoencoder . . . . .	10

	ix
2.4.1 Swap-VAE . . . . .	10
2.4.2 Swap-VICReg . . . . .	11
2.5 Swap Autoencoder . . . . .	12
2.5.1 Encoder, Decoder and Auxiliary Neural Networks . . . . .	12
2.5.2 Loss Function . . . . .	15
2.5.2.1 Results . . . . .	19
2.6 Swap Autoencoder for Multiple Subjects . . . . .	22
2.6.1 Adapted Swap Mechanism . . . . .	26
2.6.2 Loss Function . . . . .	27
2.6.3 Results . . . . .	28
2.7 Discussion . . . . .	28
<b>3 Neural Speech Decoding with Natural Voice Conversion</b>	<b>31</b>
3.1 Introduction . . . . .	31
3.2 Audio Dataset for Speech-to-Speech . . . . .	32
3.3 Speech Resynthesis . . . . .	32
3.3.1 Speech Encoder . . . . .	34
3.3.1.1 Speech to HuBERT Units Encoder . . . . .	34
3.3.1.2 F0 Quantizer . . . . .	34
3.3.1.3 Speaker Embedder . . . . .	36
3.3.2 Speech Synthesizer . . . . .	37
3.4 Neural to Speech Synthesis . . . . .	39
3.5 Discussion and Future Work . . . . .	41
<b>4 Deep Visual Feature-based Brain Decoding</b>	<b>42</b>
4.1 Introduction . . . . .	42

	x
4.2 Methods . . . . .	43
4.2.1 Dataset . . . . .	44
4.2.2 Framework . . . . .	44
4.2.3 Post-hoc classification test . . . . .	44
4.3 Results . . . . .	45
4.3.1 Post-hoc classification test . . . . .	45
4.3.2 Analysis of the Visual Cortex Regions . . . . .	45
4.4 Conclusion and Discussion . . . . .	48
<b>5 Conclusions and Future Work</b>	<b>49</b>

# List of Figures

1.1	Overview of the SwapVAE model . . . . .	3
1.2	Speech Resynthesis Process Overview . . . . .	4
2.1	Structure of Grid-Free Swin Transformer . . . . .	9
2.2	Basic Swapping Mechanism . . . . .	10
2.3	t-SNE Visualization Results for Swap-VICReg: The figure displays the content latent space, with colors corresponding to word labels for each ECoG signal. . . . .	13
2.4	The proposed disentangled neural representational learning framework	13
2.5	<b>Decoder and Auxiliary Classifier Structure</b> A. Structure of Decoder. B. Structure of Semantic classifier C. Structure of Triplet Classifier . . . . .	14
2.6	Neural Speech Decoding Framework[8] . . . . .	20
2.7	Performance Comparison of different models with PCC and STOI+	21
2.8	Performance Comparison of different models with PCC and STOI+ with Boxplot . . . . .	23

2.9	Disentangling the speech-related neural activities by visualizing Training Content and Style latent with different semantic and temporal resolution . . . . .	24
2.10	Adapted SwapAE for Enhanced Multisubject Learning . . . . .	25
3.1	Overview of the Pipeline: (a) Speech-to-Speech Training Stage: Incorporates pre-trained models—HuBERT for generating speech units, F0-Quantizer for F0 unit quantization using F0 extracted from original speech via YAAPT algorithm, and Speaker Embedder for extracting direct speaker embeddings from speech. The HifiGAN synthesizer is trained to convert these extracted latents into audible speech. (b) Neural-to-Speech Training Stage: Involves training a Neural decoder to decode HuBERT and quantized F0 units. Speaker embeddings are generated from any proxy speech using ECAPA-TDNN, enabling effective voice conversion. . . . .	33
3.2	Input to hifi-GAN: The process illustrated in the diagram involves transforming speech into a format suitable for hifi-GAN, a high-fidelity generative adversarial network used for speech synthesis. Initially, speech latent vectors are obtained. These vectors are then utilized to retrieve continuous HuBERT features and fundamental frequency (F0) features through a lookup table. These two sets of features are concatenated to form a unified feature set. Subsequently, speaker-specific embeddings are appended to each frame of the concatenated features to incorporate speaker identity into the synthesis process. This enriched feature set serves as the input for hifi-GAN, facilitating the generation of high-quality synthetic speech. . . . .	37

3.3	Detailed Architecture of the Neural Decoder: The input, denoted as $T \times E$ , passes through either an RNN or SWin Transformer, followed by Spatial MaxPooling and an MLP. The signal then traverses a common latent layer and multiple Conv1d layers at varying temporal resolutions, culminating in a HUBERT unit for the final output. This design facilitates efficient decoding and translation of neural signals. . . . .	40
4.1	Pipeline Overview: Initially, visual stimuli are processed using a pre-trained deep neural network (either ResNet or DINOv2) to extract latent embeddings. These embeddings then undergo dimensionality reduction via PCA or UMAP to isolate fine-grained features. A linear regression model with Graph-Net regularization (SpaceNet) regresses these visual latent features. Subsequently, voxels of significant weights are selected for evaluation in an image classification task via thresholding. . . . .	43
4.2	(a). Average voxel weights and the mean of weight correlation coefficients across subjects for visual subregions. (b). Image Classification Accuracy (c). Comparative Analysis of Weight Maps Across Methods: Average normalized values from the weight maps of each method across all subjects. . . . .	46

# List of Tables

3.1	Speech resynthesis results on patient speech data . . . . .	39
3.2	Speech Decode Result on sEEG Data: This table presents the correlation coefficient (CC), short-time objective intelligibility (STOI), character error rate (CER), and word error rate (WER) for each model	40
3.3	Speech Latent Prediction Accuracy on sEEG Data . . . . .	41

# Chapter 1

## Introduction

### 1.1 Background

The field of Brain-Computer Interfaces (BCIs) is at the forefront of neuroscience and technology, focusing on developing neural speech prostheses to assist individuals with speech impairments. Research utilizing electroencephalographic (ECoG) recordings has demonstrated promising results in decoding speech directly from brain activity [39]. However, creating effective decoders using machine learning presents significant challenges. The primary obstacle is acquiring ample training data, which is hindered by the logistical complexities and high costs associated with clinical experiments, coupled with the scarcity of corresponding neural and speech data [8]. Moreover, comprehending the complex relationship between ECoG and audio signals is crucial for advancing this technology [27].

Exploring the neural mechanisms underlying speech production provides essential insights into the brain's intricate processing capabilities. A major goal in neuroscience is to elucidate the roles of different brain regions and neuron popula-



tions, particularly how they collaborate to encode various inputs. Through the use of ECoG electrodes, researchers have been able to uncover latent representations, shedding light on the workings of neural circuits as they process auditory inputs and facilitate speech reproduction [44]. Advances in unsupervised learning techniques have further enhanced our understanding of the robustness and variability of neural responses, as well as the dynamic remapping that occurs as neurons learn new tasks [1].

Additionally, investigations into BCI applications in speech production have shown that both machine learning and human learning play pivotal roles in optimizing control over imagined-speech BCI systems. This research emphasizes the importance of understanding the spatial and frequency tuning of neural activity, which is essential for improving BCI control [4].

## 1.2 Related Work

### 1.2.1 SwapVAE

SwapVAE, developed by Liu et al., is a novel unsupervised learning framework where the latent space within a Variational AutoEncoder (VAE) is manually divided into content and style components [21]. This framework facilitates signal reconstruction by utilizing brain signals from rhesus macaques, enabling the disentanglement of latent features by swapping the content between two augmented views. This alignment reflects the physical attributes of movement direction and speed, as

illustrated by the following equation:

$$\min_{f,g} \sum_{i=1,2} \underbrace{\mathcal{L}_{\text{rec}}(i, g(i))}_{\text{Reconstruction loss}} + \beta \sum_{i=1,2} \underbrace{D_{KL}(i^{(s)} || i_{\text{prior}}^{(s)})}_{\text{Regularization - style space}} + \alpha \underbrace{\mathcal{L}_{\text{align}}(1^{(c)}, 2^{(c)})}_{\text{Alignment - content space}}, \quad (1.1)$$

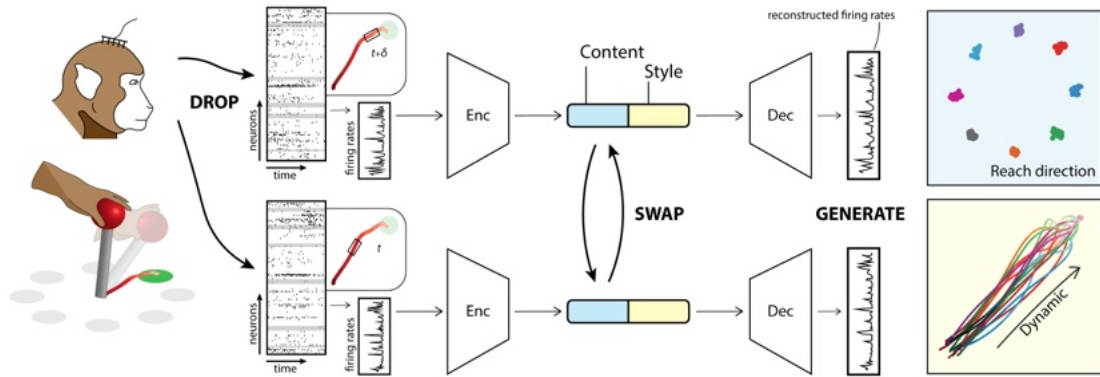


Figure 1.1: Overview of the SwapVAE model

## 1.2.2 Speech Resynthesis

The current advancements in unsupervised and quantized latent spaces have significantly enhanced speech synthesis capabilities [17]. A method designed by Polyak leverages these advancements to resynthesize natural speech using disentangled speech latents that include fundamental frequency, speech features, and speaker embeddings. This process enables straightforward voice conversion by substituting the speaker embedding, as depicted below:

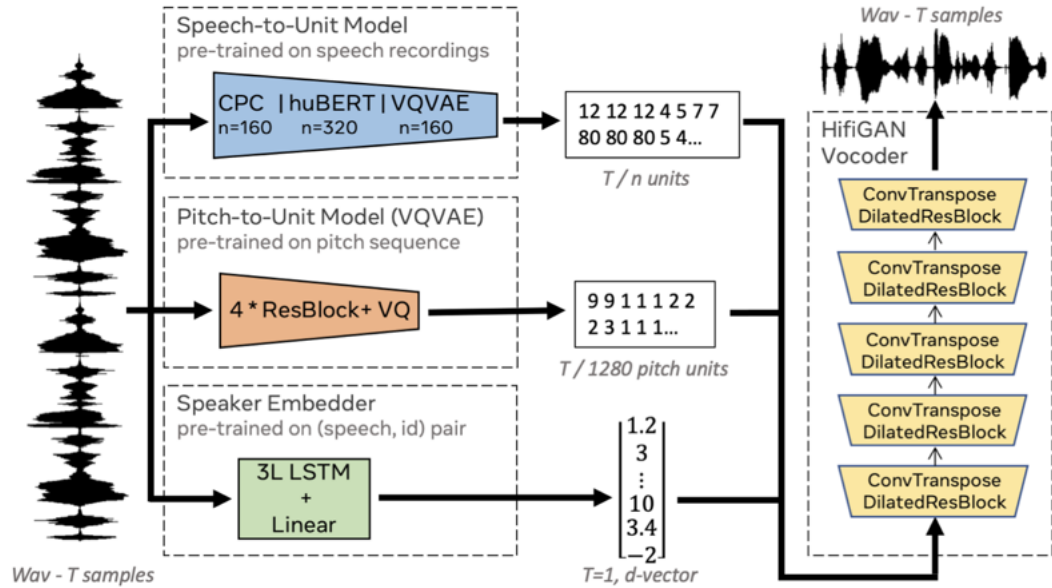


Figure 1.2: Speech Resynthesis Process Overview

### 1.2.3 ECoG to Text Decoding

Continuing exploration into the interface between neural signals and computational models, Metzger et al. have made significant strides with their ECoG-to-text decoding research, showcasing the potential to directly translate brain activity into textual outputs [27].

### 1.2.4 fMRI Decoding Model

Innovative fMRI decoding techniques, such as those discussed in "Mind's Eyes" and "Brain Decodes Deep Nets," utilize deep learning to interpret and visualize cognitive processes involved in viewing complex visual stimuli [34, 41]. These models offer profound insights into the neural representation and processing of visual information, further bridging the gap between neural activity and perceptual

experiences.

This thesis aims to highlight the significant advances in the BCI field, particularly those enhancing and expanding speech processing capabilities. Subsequent chapters will delve deeper into the methodologies used in this research, exploring both the theoretical frameworks and practical applications of these innovative technologies.

## Chapter 2

# Disentangled Neural Speech Representational Learning

### 2.1 Introduction

Understanding the intricate relationship between neural activity and speech production can offer profound insights into the brain's complex processing mechanisms. However, linking neural dynamics to specific speech features remains a formidable challenge. This study introduces an innovative framework for deriving disentangled and semantically meaningful neural representations. Our approach leverages contrastive learning, employing either a transformer or a resnet as the encoder and utilizing temporal convolutional layers for decoding. The resulting latent space effectively captures both content and instance-specific information. This concept of disentangled representation draws inspiration from techniques in computer vision that separate images into content and style components. In computer vision, content captures the essence of what an image depicts, while style

varies with each image, influencing its realism. This work[21] adapts this concept to analyze brain states by isolating content to reflect target locations and style to represent movement dynamics. Similarly, our joint neural-speech embedding is designed to segment into two parts: content that holds semantic information and an instance component that captures varying dynamics, facilitated by a novel swapping technique.<sup>1</sup>

## 2.2 Encoder Backbone

**3D ResNet** The 3D ResNet, a deep learning model, is used as a backbone to extract features from ECoG signals due to its ability to handle complex spatiotemporal data. This makes it ideal for interpreting the spatial and temporal dimensions of brain activity recorded in ECoG, which is important for applications such as brain-computer interfaces or medical diagnostics. The residual learning approach in 3D ResNet allows for training deeper networks, overcoming the challenges of vanishing gradients, which is essential for learning intricate patterns in ECoG data. Moreover, 3D ResNet is resilient to noise and variability in signals, making it a suitable choice for various neuroscientific and clinical contexts. Additionally, pre-trained models offer a practical solution in scenarios with limited ECoG data availability, enhancing their effectiveness and adaptability.

**Grid-Free Swin Transformer** Indeed, the 3D ResNet has demonstrated superior effectiveness over attention mechanism-based neural networks in the context of neural activity speech decoding, as evidenced by Chen et al. [8]. Nonetheless,

---

<sup>1</sup>This is a joint work with Xupeng Chen and Junbo Chen. I run the experiments in this project.

convolution-based neural networks offer greater flexibility and extensibility. A notable advantage is that these networks do not require the input ECoG signal to be in a grid-like format. This flexibility allows for the incorporation of a broader range of neural data types, such as sEEG and ECoG with depth electrodes, thereby expanding the dataset scale—a crucial factor for effective deep neural network training.

Integrating prior knowledge about specific regions as additional token inputs can enable the transformer model to achieve performance comparable to that of the ResNet [7], as illustrated in Figure 2.1. Consequently, we have chosen to primarily employ this approach as our backbone encoder for feature extraction.

In our design, the Swin Transformer encoder comprises a series of Swin Transformer blocks. Each block consists of a multi-head self-attention layer, a feed-forward network, and a residual connection. The multi-head self-attention layer is instrumental in allowing the model to concurrently focus on different segments of the input sequence. The feed-forward network facilitates the modeling of complex, non-linear relationships between input features and output predictions. The inclusion of a residual connection is key to preserving stability throughout the training process. A major advantage of the Swin Transformer lies in its versatility to process inputs of various spatial configurations, making it particularly adept at learning from non-grid-like electrode arrays. In our implementation, we successfully achieved a latent embedding that reduced the temporal dimension by a factor of 16 while retaining the original electrode dimensionality.

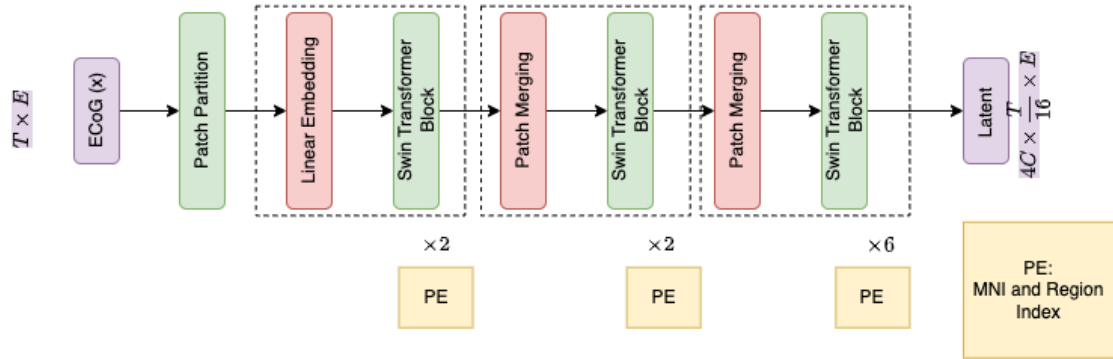


Figure 2.1: Structure of Grid-Free Swin Transformer

## 2.3 Swap Mechanism

In Liu’s study [21], a swapping mechanism was employed to create latent embeddings that carry a specific physical meaning. This was achieved by dividing the latent embedding into two parts and swapping the main informative part. In order to obtain a meaningful latent embedding, we similarly divide our latent embedding into two parts: content and style. In our case, we assume that the ”content” corresponds to the words spoken by the speaker, while the ”style” corresponds to the rest of the speech style, such as their loudness. Then, we similarly swap the ”content” parts from two different trials and do both swap reconstruction and reconstruction. To help the model learn the invariant for the content latent, two different trials selected here correspond to the same word label for the produced speech.

Here we have the basic loss function for our swapping mechanism:

$$\mathcal{L}_{recon} = \sum \|ECoG_i - ECoG_{i(Recon)}\|^2 + \sum \|ECoG_i - ECoG_{i(SwapRecon)}\|^2 \quad (2.1)$$



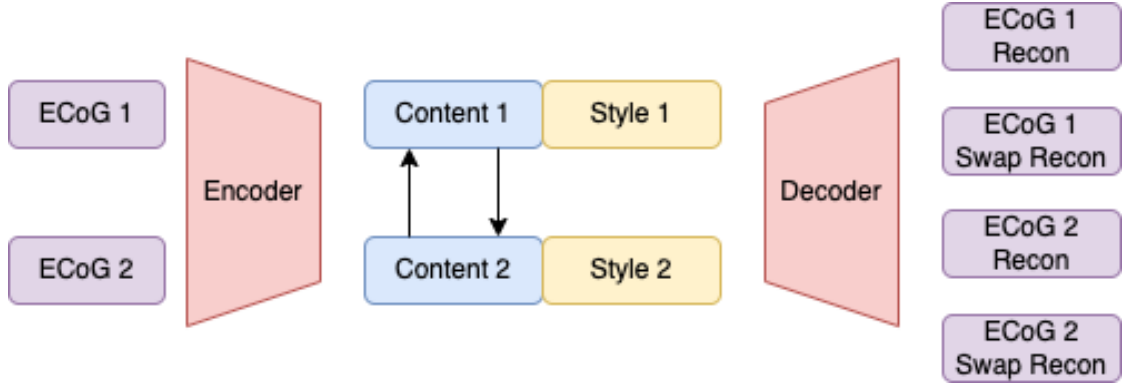


Figure 2.2: Basic Swapping Mechanism

where  $x_{i_{recon}}$  denotes the reconstructed neural activity from original latent,  $x_{i_{swap}}$  denotes the reconstructed neural activity from the swapped latent.

## 2.4 Unsupervised Swap Autoencoder

To adhere to the unsupervised learning approach, we obtain *ECoG2* by applying augmentation to the input ECoG signal. Thus, we have  $ECoG1 = ECoG$  and  $ECoG2 = f(ECoG)$ , where  $f$  represents a combination of augmentation functions discussed previously. Based on this, we explored various swapping mechanism-based models.

### 2.4.1 Swap-VAE

Our model employs a swap reconstruction loss to regulate the "content" component. However, the "style" component lacks constraints to render it meaningful. To counteract this, we use the KL divergence to regulate the "style" component, aiding in overfitting prevention and ensuring a continuous, well-structured latent space. It compels the "style" component to follow a specific distribution, such as

the Gaussian distribution. A commonly used SSL method is also implemented to enhance model generalizability. The loss function is thus defined as:

$$\mathcal{L} = \mathcal{L}_{recon} + \lambda \mathcal{L}_{KL}, \quad (2.2)$$

where  $\lambda$  is the weight assigned to the KL divergence.

**Results:** In dealing with unsupervised learning, particularly with complex inputs like ECoG signals, it is crucial to prevent feature collapse. Our experiments revealed that employing solely the Basic SwapVAE loss led to a content part embedding with similar distributions across various channels, signifying minimal informational content and thus being unsuitable for our purposes.

### 2.4.2 Swap-VICReg

Bardes et al. [3] introduced a regularization approach for self-learned features using three terms: Variance, Covariance, and Invariance. The variance term ensures sufficient representational variability and prevents trivial solutions. The invariance term fosters learning features invariant to input data transformations, often achieved by minimizing the distance between representations of different augmented inputs. Covariance Regularization encourages effective utilization of all dimensions, preventing redundancy. These terms collectively aid in model

robustness. The loss function is given by:

$$\mathcal{L}_{VIC} = \lambda \cdot \text{VarLoss} + \mu \cdot \text{InvLoss} + \nu \cdot \text{CovLoss}, \quad (2.3)$$

$$\text{VarLoss} = \frac{1}{D} \sum_{i=1}^D \max(0, \gamma - \sqrt{\text{Var}(z_i)}), \quad (2.4)$$

$$\text{InvLoss} = \frac{1}{N} \sum_{j=1}^N \|z_j - z'_j\|^2, \quad (2.5)$$

$$\text{CovLoss} = \frac{1}{D} \sum_{i \neq j} \text{Cov}^2(z_i, z_j). \quad (2.6)$$

Consequently, the overall loss function for Swap-VICReg is:

$$\mathcal{L} = \mathcal{L}_{recon} + \mathcal{L}_{VIC}. \quad (2.7)$$

**Results:** Implementing VIC in our model did not yield an interpretable speech-related latent representation (see Figure 2.3). The t-SNE visualization indicates that content features do not correlate with speech words, likely due to the lack of guidance in relating content features to speech.

## 2.5 Swap Autoencoder

### 2.5.1 Encoder, Decoder and Auxiliary Neural Networks

**Encoder** Here, we take the Swin Transformer we mentioned above, which allows us to extract better latent embedding via the multi-head attention mechanism and the positional encoding.

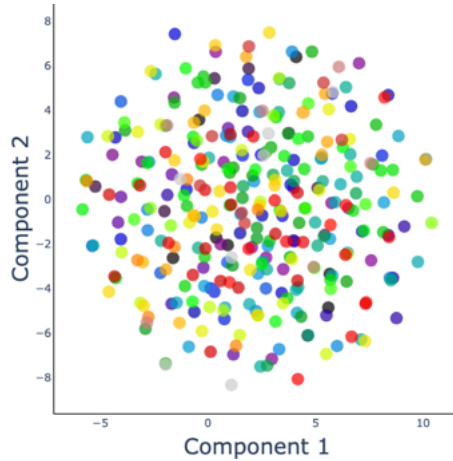


Figure 2.3: t-SNE Visualization Results for Swap-VICReg: The figure displays the content latent space, with colors corresponding to word labels for each ECoG signal.

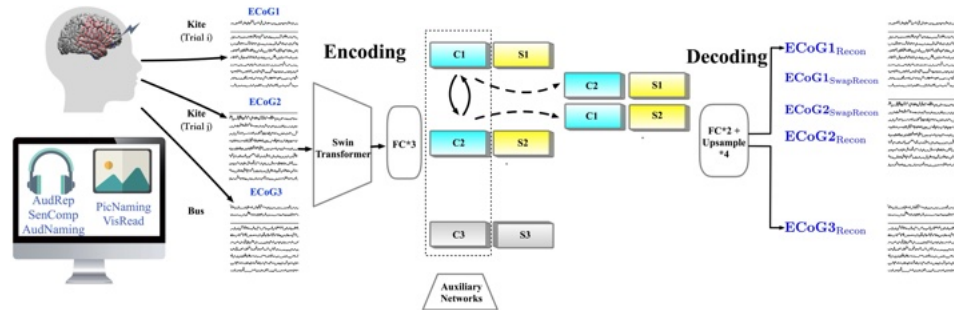


Figure 2.4: | **The proposed disentangled neural representational learning framework.** Here, we present a joint and disentangled neural latent representational learning framework. Participants are required to perform speech-related tasks and produce overt speech. The speech-paired ECoG signals are augmented. Neural activities with the same words are treated as positive pairs (ECoG1 and ECoG2), while neural activities with different words are treated as negative pairs (ECoG1/ECoG2 and ECoG3). The ECoG signals are encoded into a lower-dimensional embedding using the Swin Transformer. The latent embedding is split into two parts: content and style. The content of the two latents is then swapped to obtain a swapped latent representation. The original and swapped latents are passed through the decoder to reconstruct the original ECoG signal. Reconstruction losses, similarity losses, and auxiliary losses are used to help the model learn meaningful latent representations. The latent representation can be used to reconstruct the original ECoG signal and for downstream neural speech decoding tasks. The analysis revealed that the latent representation learned is closely related to speech semantics and speaking dynamics.

**Decoder** The decoder is comprised of a stack of temporal transposed convolutions as shown in. A group normalization layer and a leaky ReLU activation function follow each temporal transposed convolution. The temporal transposed convolutions allow the model to upsample the latent embedding back to the original temporal dimension.

**Semantic Classifier** The semantic classifier, denoted in Fig.2.5B, is responsible for learning the assignments of word labels from the content features trained using a cross-entropy loss function.

**Triplet Classifier** The triplet classifier labeled as Fig.2.5C is trained to classify whether or not two content features are positive pairs.

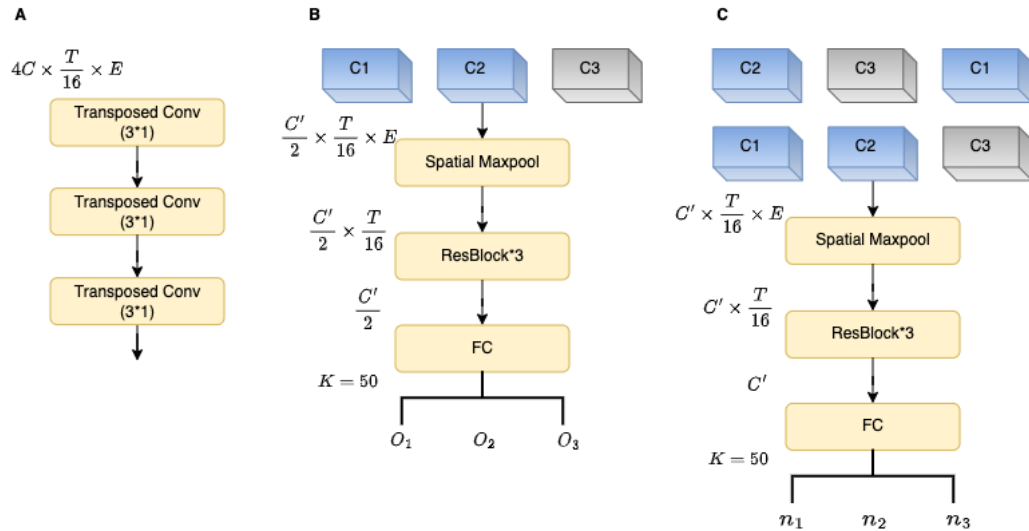


Figure 2.5: **Decoder and Auxiliary Classifier Structure** A. Structure of Decoder. B. Structure of Semantic classifier C. Structure of Triplet Classifier

## 2.5.2 Loss Function

It is difficult for a model to learn an out-of-modality representation with unsupervised learning. We tried to add some prior constraints to the model. In this case, we introduce a grouped input to our model to help it correlate to the speech information similar to work [12]. Here is how the grouped ECoG is defined:

- ECoG1: Anchor neural activity corresponding to the word "A".
- ECoG2: Different neural activity but also corresponds to the word "A".
- ECoG3: Different neural activity with different word correspondence.

Therefore, [ECoG1, ECoG2] is considered as positive pair since they represent the same word. [ECoG1, ECoG3] and [ECoG2, ECoG3] are considered two negative pairs as they correspond to two different word labels.

**Reconstruction Loss** Based on our basic swap mechanism, we also apply the reconstruction task to our third ECoG Signal here. As shown in Fig.2.4,  $ECoG_i$ ,  $i = 1, 2, 3$  are ECoG signals after the augmentation, which are fed to the Encoder. Five ECoG signals are reconstructed from latent space by the decoder.  $ECoG_{i(Recon)}$  is decoded from  $[C_i, S_i]$ ,  $ECoG_{1(SwapRecon)}$  is decoded from  $[C_2, S_1]$  and  $ECoG_{2(SwapRecon)}$  is decoded from  $[C_1, S_2]$ . Thus, the reconstruction loss here is formally defined as:

$$\mathcal{L}_{recon} = \sum_{i=1}^3 \|ECoG_i - ECoG_{i(Recon)}\|^2 + \sum_{i=1}^2 \|ECoG_i - ECoG_{i(SwapRecon)}\|^2 \quad (2.8)$$

where  $x_{i_{recon}}$  denotes the reconstructed neural activity from original latent,  $x_{i_{swap}}$  denotes the reconstructed neural activity from the swapped latent.

**Triplet Loss** Triplet Loss Methodology: As part of our strategy to effectively distinguish between positive and negative samples, we employ the Triplet Loss technique. This method directs the neural network towards precise classification in a structure that includes both positive and negative samples grouped as triplets. Our approach aims to make the data from positive ECoG pairs more alike while simultaneously distancing the data from negative pairs. The formal definition of the triplet loss is as follows:

$$\mathcal{L}_{triplet} = \|c_1 - c_2\|^2 - \|c_1 - c_3\|^2 - \|c_2 - c_3\|^2 \quad (2.9)$$

**Enhanced Content Latent with Variance-Covariance Loss** We enhance the capacity of the content latent to convey information by incorporating a variance-covariance loss, as delineated in [3]. This technique is employed to broaden the range of content representations our model can capture.

Consider a batch  $C = [c_1, \dots, c_n]$ , consisting of  $n$  content vectors each of dimension  $d$ . For each dimension  $j$ , we construct a vector  $c^j$  that comprises the  $j$ -th element from all vectors in  $C$ . The variance regularization term  $v$  is then defined as the average, over all dimensions, of a hinge function applied to the regularized standard deviation along the batch dimension. This is mathematically expressed as:

$$v(C) = \frac{1}{d} \sum_{j=1}^d \max(0, \gamma - S(c^j, \epsilon)), \quad (2.10)$$

where  $S$ , the regularized standard deviation, is given by:

$$S(c^j, \epsilon) = \sqrt{\text{Var}(c^j) + \epsilon}, \quad (2.11)$$

and the variance  $\text{Var}(c^j)$  is calculated as:

$$\text{Var}(c^j) = \frac{1}{n-1} \sum_{i=1}^n (c_i^j - \bar{c}^j)^2, \quad \text{with} \quad \bar{c}^j = \frac{1}{n} \sum_{i=1}^n c_i^j. \quad (2.12)$$

In this formulation,  $\gamma$  represents a predefined standard deviation target, set to 1 in [3], while  $\epsilon$  is a small scalar added to prevent numerical issues. This approach encourages the variance within a batch to match  $\gamma$  across each dimension, thereby averting mode collapse.

The covariance matrix for  $C$  is defined as:

$$\text{Cov}(C) = \frac{1}{n-1} \sum_{i=1}^n (c_i - \bar{c})(c_i - \bar{c})^T, \quad \text{with} \quad \bar{c} = \frac{1}{n} \sum_{i=1}^n c_i. \quad (2.13)$$

We then define the covariance regularization term  $cov$  as the sum of the squared off-diagonal elements of  $\text{Cov}(C)$ , normalized by the dimension  $d$ :

$$cov(C) = \frac{1}{d} \sum_{i \neq j} [\text{Cov}(C)]_{i,j}^2. \quad (2.14)$$

This term aims to minimize the off-diagonal elements of  $\text{Cov}(C)$ , thereby decorrelating the different dimensions of the embeddings and ensuring they don't encode redundant information. Such decorrelation at the embedding level translates into decorrelation at the representation level.



Finally, the variance-covariance loss is formulated as:

$$\mathcal{L}_{VC} = v(C) + cov(C) \quad (2.15)$$

**Utilizing Cross-Entropy in Classification** In our approach, cross-entropy serves as the loss function for two auxiliary classifiers that aid in the learning of content features. These classifiers are depicted in Fig. 2.5 parts c and d. The first classifier, a semantic classifier, processes content inputs  $c_1, c_2, c_3$  and produces outputs  $o_1, o_2, o_3$ , each classified into one of 50 categories. The second classifier, known as the triplet classifier, assesses whether pairs from  $c_i, i = 1, 2, 3$  form positive or negative pairs, yielding predictions  $n_1, n_2, n_3$  for pairs  $[c_1, c_2]$ ,  $[c_2, c_3]$ , and  $[c_1, c_3]$ , respectively.

The loss function for the semantic classifier is formulated as follows:

$$\mathcal{L}_{CE_{semantic}} = - \sum_i^3 \sum_{k=1}^{50} y_{o_i,k} \log(p_{o_i,k}) \quad (2.16)$$

In the case of the triplet classifier, the loss function is defined by:

$$\mathcal{L}_{CE_{triplet}} = - \sum_i^3 \sum_{k=1}^2 y_{n_i,k} \log(p_{n_i,k}) \quad (2.17)$$

Here,  $y$  denotes the one-hot vector representing the true label, and  $p$  indicates the softmax output of the classifier. For semantic classification, there are 50 classes ( $K=50$ ), and for triplet classification, there are two possible outcomes, positive or negative pairings ( $K=2$ ).

By combining these loss functions, we derive the overall cross-entropy loss for

classification:

$$\mathcal{L}_{CE} = \mathcal{L}_{CE_{semantic}} + \mathcal{L}_{CE_{triplet}} \quad (2.18)$$

The comprehensive loss function for our model, therefore, is:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{recon} + \lambda_2 \mathcal{L}_{triplet} + \lambda_3 \mathcal{L}_{\mathcal{V}\mathcal{C}} + \lambda_4 \mathcal{L}_{CE}, \quad (2.19)$$

where each  $\lambda_i$  represents the weighting factor for its corresponding loss component.

In our configuration, we have set  $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 1$ .

### 2.5.2.1 Results

We employed our speech decoding framework as shown in Fig.2.6 across N=48 participants who consented to complete a series of speech tasks. The participants were undergoing treatment for refractory Epilepsy with implanted electrodes for their clinical care. During the hospital stay, we acquired synchronized neural and acoustic speech data. ECoG data were obtained from five participants with hybrid-density(HB) sampling (clinical-research grid) and 43 participants with low-density(LD) sampling (standard clinical grid), who took part in five speech tasks: Auditory Repetition (AR), Auditory Naming (AN), Sentence Completion (SC), Word Reading (WR), and Picture Naming (PN). These tasks were designed to elicit the same set of spoken words across tasks while varying the stimulus modality. We provided 50 repeated unique words (400 total trials per participant), all of which were analyzed locked to the onset of speech production. We trained a model for each participant using 80% of the available data for this participant. We evaluated the model on the remaining 20% of data (with the exception of the more stringent word-level cross-validation).

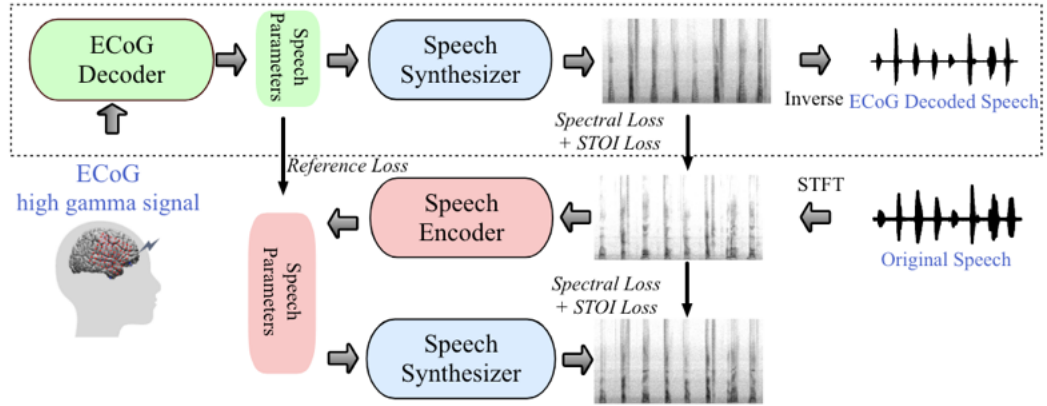


Figure 2.6: Neural Speech Decoding Framework[8]

We compare the performance of neural speech decoding to evaluate the decoding ability of the latent representation from Swap pretrained model. We use the evaluation metrics CC and STOI+ and show the results in 2.7. We compare the baseline ResNet model used in the first chapter and the Swin Transformer model without pre-training. We show a consistent increase in performance for both metrics (in Fig. 2.7a and d). We then use the Swap framework as pretraining and fine-tune the Swin Transformer in a neural speech decoding task. We show that the Swap pre-trained Swin Transformer using only grid electrodes has further gain in terms of CC and STOI+ in most of the participants (in Fig. 2.7b and e). Further comparison shows that the Swap pre-trained Swin Transformer with all electrodes used could have an even greater gain in performance. This indicates the advantage of our Swap-pretrained framework in finding a better latent semantic representation for downstream decoding.

Here, we compare the ResNet baseline, SwinT from scratch, Swap pretrained SwinT with grid electrodes, and Swap pretrained SwinT with all electrodes in the boxplot in Fig. 2.8. Consistent gains are observed in both CC and STOI+

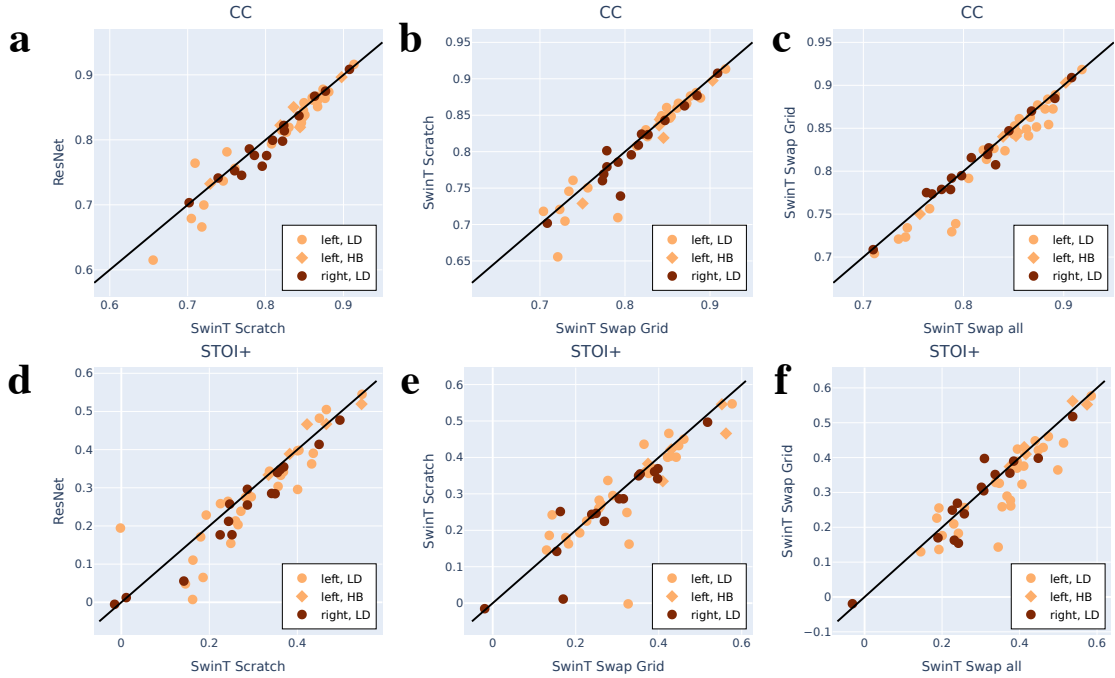


Figure 2.7: | **Performance Comparison of different models with PCC and STOI+** We compare the performance of neural speech decoding to evaluate the decoding ability of the latent representation. The evaluation metrics we use are CC and STOI+. We first compare the baseline ResNet model used in the first chapter and the Swin Transformer model without pre-training. We show a consistent increase in performance for both metrics (in **a** and **d**). We then use the Swap framework as pretraining and fine-tune the Swin Transformer in a neural speech decoding task. We show that the Swap pre-trained Swin Transformer using only grid electrodes has further gain in terms of CC and STOI+ in most of the participants (in **b** and **e**). Further comparison shows that the Swap pre-trained Swin Transformer with all electrodes used could have an even greater gain in performance. This indicates the advantage of our Swap-pretrained framework in finding a better latent semantic representation for downstream decoding tasks.

plot metrics. Each point in the plot represents one participant. The example brains show the grid-only electrodes and all electrodes. The latter contains grid and strip and/or depth electrodes. We can see that the Swap pretrained SwinT models outperform the other models on both metrics. This suggests that the Swap pretraining procedure is effective in improving the performance of SwinT on ECoG to speech decoding. The biggest gain comes from the all-electrodes model, suggesting that the Swap pretraining procedure equipped with SwinT can learn a better latent representation than previous arts.

To understand the latent representation, we apply t-SNE[23] to the latent content and style features formed by our model and visualize them in 2D. Fig. 2.9a shows the content latent clustered into 50 classes that correspond exactly to the word of the overt utterance. Fig. 2.9 c shows that the style representations are clustered according to temporal dynamics. This indicates that speech dynamics may contain information other than the semantic information encoded in the content, and our model disentangles this information. However, these are visualization for the training samples. However, it couldn't generalize well on test data which indicates the model overfits on the training data due to our limited data. In this case, our proposed method is effective for neural speech joint embedding disentangling to some extent.

## 2.6 Swap Autoencoder for Multiple Subjects

To mitigate the overfitting challenge in our model, we have expanded the scope of our SwapAE framework to encompass multiple patients, as illustrated in Fig.2.10. This strategic enhancement is designed to diversify the training data, thereby

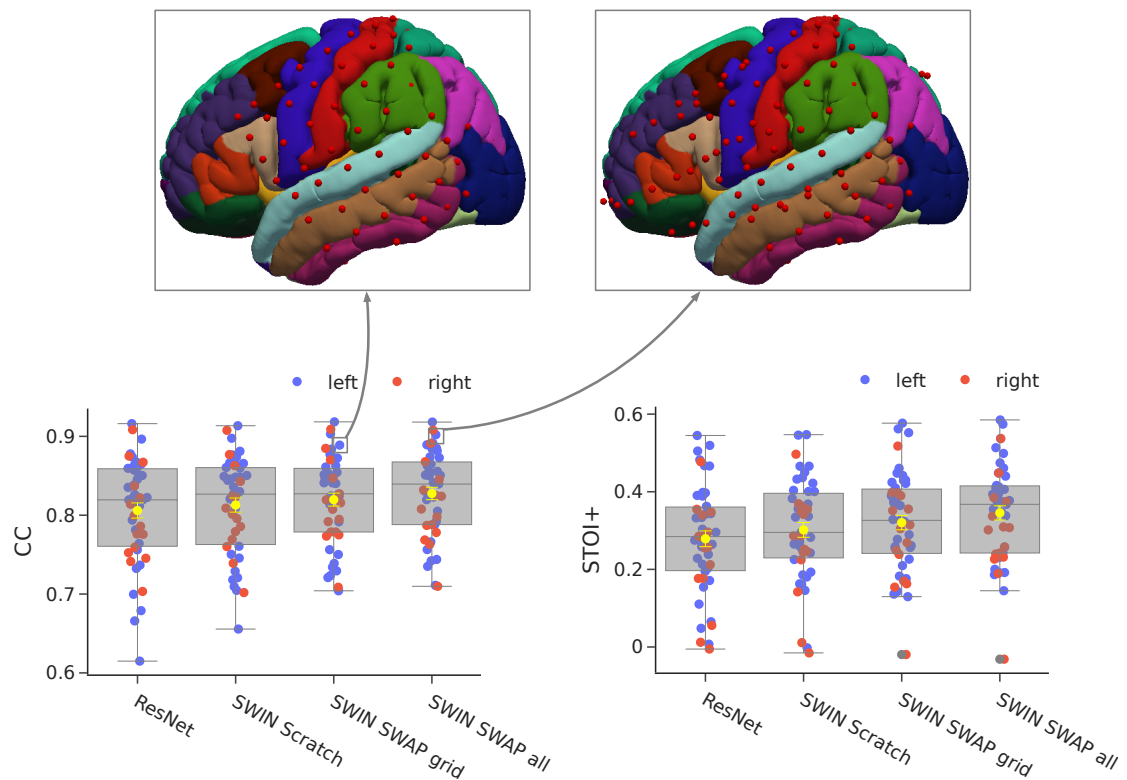


Figure 2.8: | **Performance Comparison of different models with PCC and STOI+ with Boxplot** Here we compare the ResNet baseline, SwinT from scratch, Swap pretrained SwinT with grid electrodes, and Swap pretrained SwinT with all electrodes. Consistent gains are observed in both CC and STOI+ plot metrics. Each point in the plot represents one participant. The example brains show the grid-only electrodes and all electrodes, which contain both grid electrodes and strip and/or depth electrodes.

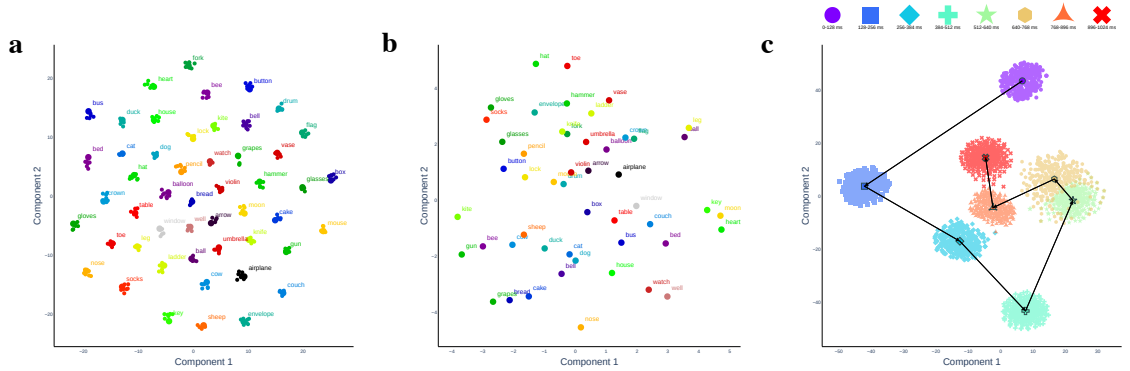


Figure 2.9: | **Disentangling speech-related neural activity by visualizing latent content and style with different semantic and temporal resolution**  
 To understand the latent representation, we apply t-SNE[23] to the latent content and style features formed by our model and visualize them in 2D. **a** shows the content latent for training samples clustered into 50 classes that correspond exactly to the word of the overt utterance. **b** shows how content latent from test samples is distributed. **c** shows that the style representations for training samples are clustered according to temporal dynamics. This indicates that speech dynamics may contain information other than the semantic information encoded in the content.

strengthening the model’s generalization capabilities.

By structuring the latent space in this nuanced manner, the model is equipped to disentangle and understand complex patterns across different patients. The content component focuses on the temporal dynamics of the data, ensuring that essential time-related information is captured. In contrast, the patient style component is tailored to spatial aspects, enabling the model to recognize and adapt to the unique electrocortical attributes of individual patients. Finally, the instance style aims to filter out and manage extraneous elements such as noise, enhancing the model’s robustness and accuracy in diverse scenarios. This comprehensive approach in the modified SwapAE framework is a strategic step towards achieving more reliable and generalizable results in multisubject data analysis.

Similar to the group setting in the previous swapAE, we set up a new group input:

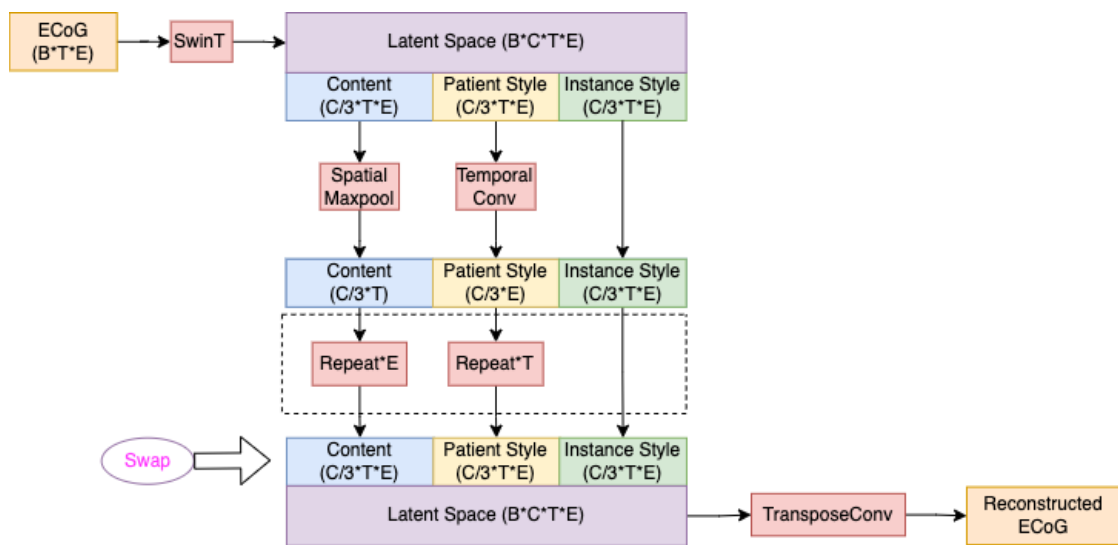


Figure 2.10: | **Adapted SwapAE for Multisubject Application:** Building on the single-patient SwapAE foundation, this version maintains the same core encoder and decoder components. The latent space is now intricately partitioned into three segments: content, patient style, and invariance style. We specifically design the content latent to encapsulate only temporal information, the patient style to uniquely reflect spatial dimensions corresponding to varied electrocortical graphical features, and the instance style to capture additional elements like noise and other variabilities.



- ECoG 1: Initial neural activity recording for Patient 1, focusing on their response to Word 1 during the first trial.
- ECoG 2: A distinct neural activity pattern from Patient 1, still in response to Word 1, but from a subsequent trial.
- ECoG 3: Different neural activity from Patient 1, now in response to a new word, Word 2.
- ECoG 4: Neural activity from Patient 2, corresponding to their reaction to Word 1.
- ECoG 5: A unique neural response from Patient 2, related to a different word, indicated as Word n (not Word 1).

### 2.6.1 Adapted Swap Mechanism

**Content Swap** To obtain a disentangled and meaningful content latent that accurately represents word labels, we implement a content swap mechanism. This involves exchanging content between different ECoG signals that correspond to identical words.

**Patient Swap** Similarly, to achieve a meaningful patient style latent representation, we swap patient styles between ECoG signals from the same patients. For example, we might swap styles between signals like ECoG1 and ECoG2, each originating from a separate patient. This approach is aimed at enhancing the distinctiveness of patient-specific characteristics in the data.

## 2.6.2 Loss Function

**Reconstruction Loss** In our enhanced model, we expand the reconstruction loss to cover 5 ECoG signals, compared to the 3 signals in our previous single version swap-AE. The loss is now formulated as follows:

$$\mathcal{L}_{recon} = \mathcal{L}_{OriginalRecon} + \mathcal{L}_{ContentRecon} + \mathcal{L}_{PatientRecon} \quad (2.20)$$

$$\mathcal{L}_{OriginalRecon} = \sum_{i=1}^5 \|ECoG_i - \phi(C_i + P_i + I_i)\|^2 \quad (2.21)$$

$$\mathcal{L}_{ContentRecon} = \sum_{i,j} \|ECoG_j - \phi(C_i + P_j + I_j)\|^2, \text{ where } i, j \in \{1, 2, 3\} \quad (2.22)$$

$$\mathcal{L}_{PatientRecon} = \sum_{i,j} \|ECoG_j - \phi(C_j + P_i + I_j)\|^2, \text{ where } i, j \in \{1, 2, 3\} \quad (2.23)$$

Where  $\phi$  denotes the decoder part of the autoencoder. Note: The sums in  $\mathcal{L}_{ContentRecon}$  and  $\mathcal{L}_{PatientRecon}$  were corrected to iterate over 5 signals for consistency. For the content reconstruction loss, the content latent extracted from  $ECoG_1$ ,  $ECoG_2$  and  $ECoG_4$  are swapped between latent from these three signals. For the patient reconstruction loss, the patient latent extracted from  $ECoG_1$ ,  $ECoG_2$  and  $ECoG_3$  are swapped. For both of these two losses, reconstruction target follows the unaltered latent parts.

**Variance Covariance Loss** To ensure that the content and patient latent spaces learn meaningful representations, we employ the Variance Covariance loss on both.

This loss is defined as:

$$\mathcal{L}_{VC_{all}} = \sum_{i=1}^5 \mathcal{L}_{VC}(C_i) + \sum_{i=1}^5 \mathcal{L}_{VC}(P_i) \quad (2.24)$$

**Triplet Loss** To foster similarity in content from the same word and patient style from the same patient, we apply triplet loss to both content and patient styles. This approach aims to enhance the distinctiveness and consistency within each category of data.

### 2.6.3 Results

For the results of this multi-patient swapVAE, the model still requires further modification to generate the desirable content part.

## 2.7 Discussion

This chapter presented a novel approach utilizing self-supervised alignment to dissect neural activity into distinct latent subspaces, providing new perspectives on the relationship between neural dynamics and speech features. Our model effectively captures individual variations across participants, suggesting its potential applicability in broader neuroscientific studies.

We employed neural data augmentation strategies such as temporal jitter, channel-wise dropout, and additive noise to enhance the model’s robustness and aid in the disentanglement of semantic content from speech dynamics. The model adeptly balances encoding of semantic and dynamic information within neural activity, which is anticipated to enhance neural speech decoding performance in

downstream tasks.

Our framework integrates a sequential backbone architecture to leverage dynamic speech aspects, incorporating these into the model's learning process. The successful use of a transformer-based architecture in conjunction with contrastive learning illustrates its capability in modeling complex latent structures over extended periods.

It is important to note that the results presented here are preliminary and represent an initial exploration into this complex area of research. The disentangled latent representations show effective clustering and semantic alignment only with training data. When the ECoG decoder is fine-tuned for downstream speech decoding tasks, this distinct clustering tends to diminish, indicating a potential overfitting to training data or a need for further model adjustments to preserve these characteristics in applied settings.

The process of disentangling these representations without explicit labels is inherently challenging, requiring careful tuning of the loss functions and thoughtful design of the model to facilitate the learning of effective latent representations. The intrinsic complexity of speech production and the potential entanglement within neural activities add to these challenges. Most current methodologies focus on single-subject analyses, and the preliminary nature of our findings underscores the necessity for further investigations. Extending this research to include multi-subject data could substantially enrich the understanding and enhance the generalizability of the models.

Future work will aim to not only decode "what" and "how" information is encoded in neural signals but also to identify "whose" information is encoded, opening new avenues for personalized medicine and customized neural interfaces. This research will continue to evolve, incorporating more data and refining method-

ologies to better understand and utilize the rich information encoded in neural activity related to speech.

## Chapter 3

# Neural Speech Decoding with Natural Voice Conversion

### 3.1 Introduction

This chapter <sup>1</sup> presents a novel approach in neural speech synthesis, focusing on the conversion of brain activity to speech while maintaining the speaker’s natural voice characteristics. The methodology integrates two advanced training stages: Speech-to-Speech and Neural-to-Speech, utilizing state-of-the-art models such as Hubert and HifiGAN for speech unit generation and synthesis.

The primary motivation for this research is to develop communication aids for individuals with speech impairments, enabling them to speak with their own voice (or their preferred proxy’s voice) characteristics. The approach leverages sophisticated neural networks and diverse datasets like VCTK and VoxCeleb to ensure the synthesized speech is both clear and personal.

---

<sup>1</sup>This is a joint work with Xupeng Chen. I setup the pipeline and run the experiments in this project.

## 3.2 Audio Dataset for Speech-to-Speech

**VCTK** The VCTK dataset [40], which includes recordings from 109 native English speakers of various accents, serves as a foundational element in our experiments. Its diversity is key to enhancing our model’s ability to handle accent variations in speech synthesis, thereby increasing robustness and adaptability. The dataset’s comprehensive annotations and broad phonetic diversity are critical for training models that can accurately replicate the nuanced characteristics of human speech. The audio sampling rate for VCTK is resampled to 16K.

**VoxCeleb** In contrast, the VoxCeleb dataset provides an extensive collection of over 100,000 utterances from 1,251 celebrities, derived from online video sources. With its variable recording conditions and background noise [29], VoxCeleb offers a challenging yet valuable real-world testing environment. It allows us to evaluate our speech synthesis model’s performance in less controlled acoustic conditions, ultimately enhancing the model’s ability to deliver clear and intelligible speech in various adverse scenarios. The sampling rate for the VoxCeleb dataset is 16K.

## 3.3 Speech Resynthesis

Similar to the model described in [33], we employ a similar encoder-decoder architecture to re-synthesize the speech. The speech encoder part contains three sub-encoders: Speech to HuBERT Units encoder, F0 Quantizer, and Speaker Embedder. Similarly, we use the Hifi-GAN as the synthesizer to synthesize the speech from the speech latent generated from three sub-encoders. The overall overview is shown in Figure 3.1a.

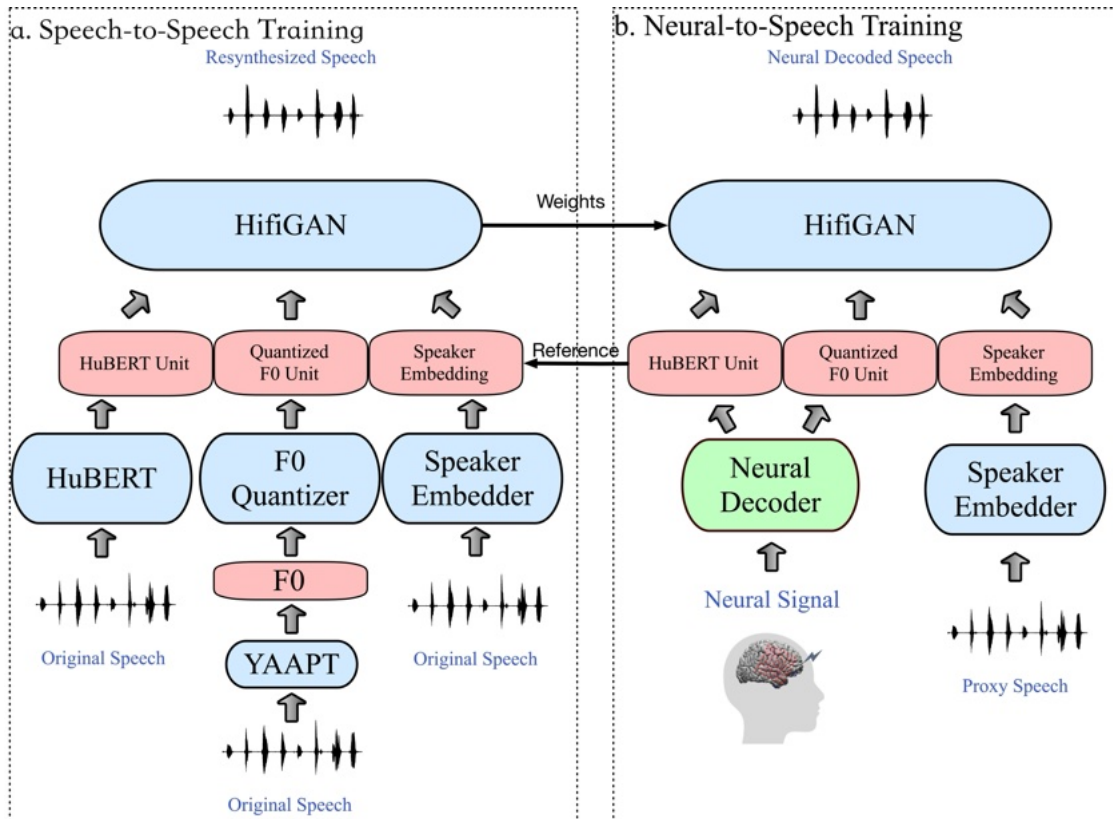


Figure 3.1: Overview of the Pipeline: (a) Speech-to-Speech Training Stage: Incorporates pre-trained models—HuBERT for generating speech units, F0-Quantizer for F0 unit quantization using F0 extracted from original speech via YAAPT algorithm, and Speaker Embedder for extracting direct speaker embeddings from speech. The HifiGAN synthesizer is trained to convert these extracted latents into audible speech. (b) Neural-to-Speech Training Stage: Involves training a Neural decoder to decode HuBERT and quantized F0 units. Speaker embeddings are generated from any proxy speech using ECAPA-TDNN, enabling effective voice conversion.



### 3.3.1 Speech Encoder

#### 3.3.1.1 Speech to HuBERT Units Encoder

The Speech to HuBERT Units Encoder leverages the HuBERT model[17], which is integral for capturing the linguistic content of speech. This encoder specifically converts raw speech into HuBERT units, serving as a crucial intermediary step in transforming speech into a format amenable for further processing. The Hubert encoder operates by first segmenting the input speech into smaller, manageable frames (20ms per frame for audio sampled at 16kHz), which are then individually analyzed to extract their acoustic properties. These properties are encoded into a discrete set of HuBERT units, effectively compressing the speech data while preserving essential linguistic features. This process not only facilitates a more compact representation of speech but also enhances the model’s responsiveness to linguistic nuances, thus improving its overall performance in speech-related tasks.

#### 3.3.1.2 F0 Quantizer

The F0 quantizer is pre-trained with the VQ-VAE framework on the F0 reconstruction task. It consists of the F0 encoder and bottleneck and is used to generate discrete units as the representation embedding of F0.

**F0 Encoder** The encoder includes four residual blocks, each performing a down-sampling operation on the F0 by a factor of two. This process not only reduces the temporal resolution but also increases the feature channels. To prepare the F0 for this encoding, the YAAPT[19] is used to extract the F0 from the original audio, where one F0 value is sampled from every 80 frames. This equates to F0 being downsampled by 80 times compared to the original audio, resulting in 200 F0

values per second. Subsequently, these are further downsampled by a factor of 16 during encoding, resulting in an F0 representation of approximately 12.5 units per second. The resultant features across all channels at each downsampled temporal point are then vector quantized.

**Codebook** The VQ-VAE codebook is learnable and stores a set of discrete embeddings. The continuous F0 embeddings from the encoder are quantized by mapping them to the nearest neighbors in this codebook, transforming them into a discrete set of indices.

**F0 Decoder** The decoder uses four transposed convolutional layers to reconstruct the F0 from the discrete embeddings retrieved from the codebook. These layers progressively upsample the quantized representations back to the original resolution of the F0.

**Loss Function** The training of the f0-VQ-VAE is directed by a loss function composed of a reconstruction term and a commitment term, formulated as follows:

$$\begin{aligned}
 L(E_{F_0}, C, D_{F_0}) &= L_{recon} + \beta L_{commit}, \\
 L_{recon}(E_{F_0}, C, D_{F_0}) &= \frac{1}{T'} \sum_{t=1}^{T'} \|p_t - D_{F_0(t)}(e_s)\|_2^2, \\
 L_{commit}(E_{F_0}, C) &= \frac{1}{L'} \sum_{s=1}^{L'} \|h_s - \text{sg}[e_{z_s}]\|_2^2,
 \end{aligned} \tag{3.1}$$

where:

- $p_t$  represents the target F0 value at time  $t$ ,
- $D_{F_0(t)}(e_s)$  denotes the reconstructed F0 from the decoder for the embedding

$e_s$  at time  $t$ , which are derived at a lower temporal resolution  $s = \frac{t}{s}$ .

- `sg` indicates the stop gradient operation, which detaches the embedding from the gradient updates during backpropagation,
- $h_s$  is the original embedding from the encoder for segment  $s$ ,
- $e_{z_s}$  is the closest codebook embedding to  $h_s$ ,
- $\beta$  is a weighting factor that balances the reconstruction and commitment loss components.

This loss function ensures the decoder can reconstruct the F0 accurately from the quantized embeddings and maintains a strong alignment with the original embeddings, minimizing the loss of information during the quantization process. The reconstructed F0 correlation coefficient can be achieved at 0.88 under this setting while testing on our patient speech dataset.

### 3.3.1.3 Speaker Embedder

The Speaker Embedder utilizes the ECAPA-TDNN[11] architecture, renowned for its effectiveness in capturing distinct speaker characteristics. This architecture employs a series of densely connected convolutional layers, which are designed to process and identify unique vocal traits from input speech signals. The extracted features undergo a temporal aggregation process, which synthesizes the information across the entire input sequence, resulting in a dense vector representation of the speaker’s voice. This representation encapsulates the unique timbral and dynamic characteristics of the speaker, allowing the synthesized speech to retain the individuality of the speaker’s voice. The inclusion of attention mechanisms

further refines the embedding process, ensuring that salient features are emphasized, thereby enhancing the model’s accuracy in speaker characterization.

### 3.3.2 Speech Synthesizer

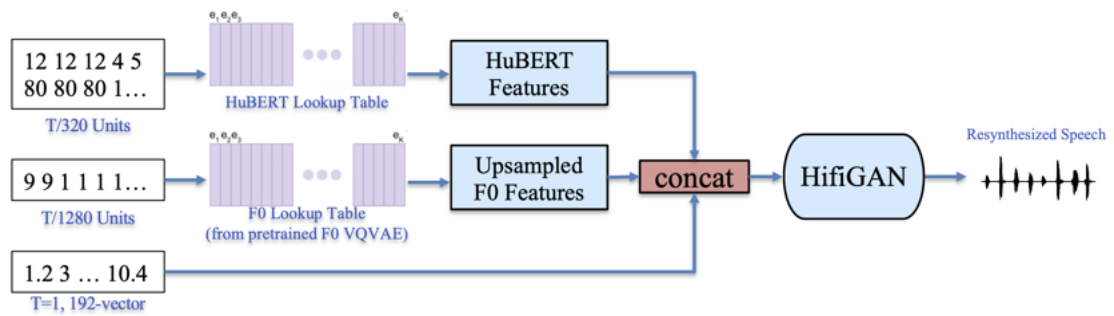


Figure 3.2: Input to hifi-GAN: The process illustrated in the diagram involves transforming speech into a format suitable for hifi-GAN, a high-fidelity generative adversarial network used for speech synthesis. Initially, speech latent vectors are obtained. These vectors are then utilized to retrieve continuous HuBERT features and fundamental frequency (F0) features through a lookup table. These two sets of features are concatenated to form a unified feature set. Subsequently, speaker-specific embeddings are appended to each frame of the concatenated features to incorporate speaker identity into the synthesis process. This enriched feature set serves as the input for hifi-GAN, facilitating the generation of high-quality synthetic speech.

The Hifi-GAN synthesizer is configured to receive inputs composed of concatenated features from the HuBERT units and F0 units where the speaker embedding is then concatenated to each frame of concatenated features shown as Fig.3.2. This arrangement ensures that the synthesized speech retains the content, prosody, and timbre of the original speech. The training of Hifi-GAN incorporates a multi-objective loss function, which includes adversarial, feature-matching, and reconstruction losses, as defined in [33]. This complex loss function helps in generating high-quality, realistic speech.

For the hifi-GAN training, we use the same loss used in the [33]:

$$\begin{aligned}
 L_G^{multi}(D, G) &= \sum_{j=1}^J (L_{adv}(G, D_j) + \lambda_{fm}L_{fm}(G, D_j)) + \lambda_r L_{recon}(G), \\
 L_D^{multi}(D, G) &= \sum_{j=1}^J L_D(G, D_j),
 \end{aligned}
 \tag{3.2}$$

where:

- $G$  denotes the generator in the Generative Adversarial Network (GAN), responsible for generating synthetic speech that mimics the real speech samples.
- $D$  represents the discriminator in the GAN, tasked with distinguishing between real and generated speech samples.
- $D_j$  denotes the  $j$ th sub-discriminator, including multi-scale discriminators and multi-period discriminators, which constitute the entire discriminator  $D$ .
- $L_{adv}$  is the adversarial loss, which measures how well  $G$  can deceive  $D$ .
- $L_D$  is the discriminator’s loss, which quantifies  $D$ ’s ability to correctly classify real and generated samples.
- $L_{fm}$  is the feature-matching loss, which ensures that the features of the generated samples closely match those of the real samples across multiple layers of  $D$ .
- $L_{recon}$  is the reconstruction loss, specifically the L1 loss on the mel-spectrogram, which measures the accuracy of the reconstructed audio.
- $\lambda_{fm}$  and  $\lambda_r$  are weighting coefficients set to 2 and 45, respectively, balancing

the influence of the feature-matching and reconstruction losses relative to the adversarial loss.

Here we set  $\lambda_{fm} = 2$  and  $\lambda_r = 45$ .

**Training Strategy** The training of this speech resynthesis framework is carefully structured. The HuBERT and speaker embedder models are employed directly from existing pre-trained models because they are well-trained on large datasets, ensuring robust initial feature extraction. However, the F0-VQ-VAE and Hifi-GAN require specific training on speech datasets. Initial training is conducted on the VCTK dataset, which, despite its diversity, is limited in scale. To enhance the model’s generalizability, further fine-tuning is performed on the VoxCeleb dataset, which contains a wide range of speakers and recording conditions. This two-stage training process significantly improves the model’s ability to synthesize speech from unseen speakers, as evidenced by the improved Mel-Spectrogram Correlation Coefficient in our evaluations.

Dataset Used	Mel-Spectrogram CC
VCTK	0.719
VCTK+VoxCeleb	0.926

Table 3.1: Speech resynthesis results on patient speech data

### 3.4 Neural to Speech Synthesis

As detailed in Figures 3.1b and 3.3, training the neural decoder to decode discrete logits of HuBERT and F0 units simplifies the task by framing it as classification rather than regression. According to [27], the highest performance in neural decoding has been achieved using an RNN-based architecture. However, RNNs are

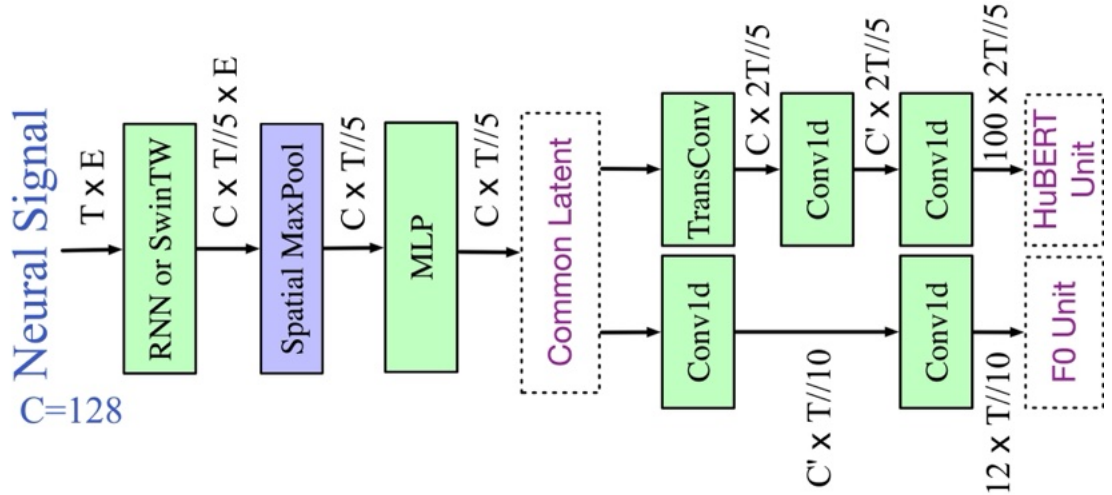


Figure 3.3: Detailed Architecture of the Neural Decoder: The input, denoted as  $T \times E$ , passes through either an RNN or SWin Transformer, followed by Spatial MaxPooling and an MLP. The signal then traverses a common latent layer and multiple Conv1d layers at varying temporal resolutions, culminating in a HUBERT unit for the final output. This design facilitates efficient decoding and translation of neural signals.

	CC	STOI	CER	WER
RNN	0.803	0.465	0.85	1.04
SwinTW	0.727	0.325	1.32	1.38

Table 3.2: Speech Decode Result on sEEG Data: This table presents the correlation coefficient (CC), short-time objective intelligibility (STOI), character error rate (CER), and word error rate (WER) for each model

limited in multi-patient scenarios due to their sensitivity to spatial variations in neural electrode placement. To overcome this, a transformer-based subject-agnostic decoding model is developed, which performs robustly across different electrode configurations [7]. Initial results are summarized in Table 3.2. From the discrete units prediction accuracy in the Table 3.3, the accuracy is very low. That's the reason for the undesirable speech decoding result. Therefore, we need to tune our neural decoder architecture further for better unit sequence prediction.

Prediction Accuracy	F0 Units	HuBERT Units
RNN	0.347	0.278
SwinTW	0.285	0.125

Table 3.3: Speech Latent Prediction Accuracy on sEEG Data

### 3.5 Discussion and Future Work

The project presents promising initial results, but there are several avenues for improvement and future exploration to enhance performance and applicability:

**Enhancing F0-VQ-VAE Performance** Current outcomes suggest that the F0-VQ-VAE may not capture enough granular details in the quantized F0 units, potentially limiting the naturalness and expressiveness of synthesized speech. Considering alternative models such as FSQ-VAE [26] could provide a more refined quantization of F0 features, potentially leading to improved speech re-synthesis quality. Therefore, a better audio guidance can be provided for our neural decoding.

**Optimizing Neural Decoders for Multi-Patient Scenarios** The present models, while effective in some respects, show limitations in decoding performance, particularly in multi-patient scenarios where spatial variability in electrode placements can affect results. Developing and testing other architectures, such as advanced transformer models or domain adaptation techniques, might yield better generalization across different patients without compromising the decoding accuracy.

By addressing these areas, the project could significantly advance the field of speech synthesis from brain activity, offering more personalized and effective communication aids for those in need.



## Chapter 4

# Deep Visual Feature-based Brain Decoding

### 4.1 Introduction

Brain decoding is an important technique for deriving insights into the brain’s functions by finding how voxel-level activation data can be used to predict certain stimuli or response variables[34, 42]. In this work, we investigate a simple tweak to the traditional classification-based decoding method: instead of using pre-defined classification labels [6, 10, 14, 15, 18], we propose to use pre-trained representations from deep neural network (DNN) models. In the interest of space, we focus on the visual cortex’s response to natural scenes from the Natural Scenes Dataset (NSD), but our method can be easily applied to other domains, such as auditory processing. While some previous works have delved into regression-based approaches [22, 28, 31, 32, 35, 36], our methodology introduces a distinctive perspective. Our proposed new decoding method removes the need for existing stimuli labels and

provides a weight map that better aligns with the underlying scene recognition process compared to classification-based decoding. Through a post-hoc classification test of scene classification, we show that our method preserves the class-related information even when not explicitly optimized for it, achieving a very similar performance as classification-based decoding. These advantages make our method a simple drop-in replacement for many decoding-style analyses involving complex responses or stimuli.<sup>1</sup>

## 4.2 Methods

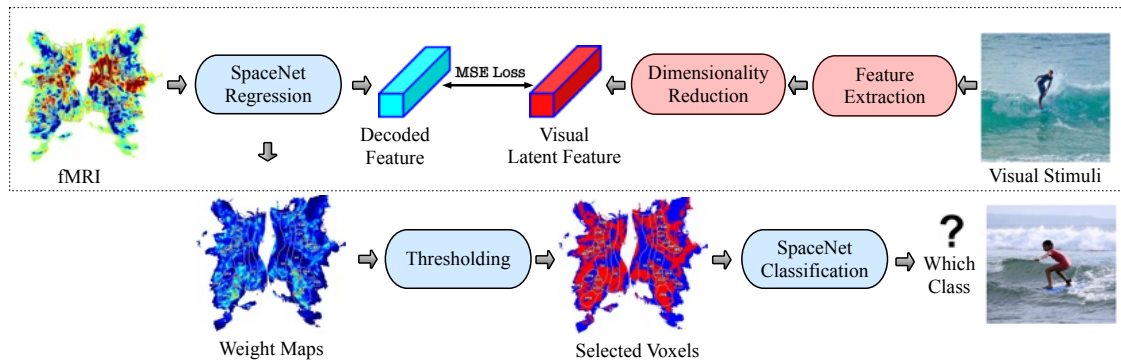


Figure 4.1: Pipeline Overview: Initially, visual stimuli are processed using a pre-trained deep neural network (either ResNet or DINOv2) to extract latent embeddings. These embeddings then undergo dimensionality reduction via PCA or UMAP to isolate fine-grained features. A linear regression model with Graph-Net regularization (SpaceNet) regresses these visual latent features. Subsequently, voxels of significant weights are selected for evaluation in an image classification task via thresholding.

<sup>1</sup>This is a joint work with Nika Emami, Chris Liu, and Xupeng Chen. I run the experiments and do brain visualization in this project.

### 4.2.1 Dataset

Our research employs the Natural Scenes Dataset (NSD)[2], a comprehensive fMRI dataset captured at 7T featuring whole-brain, high-resolution measurements from eight healthy adults. Participants were exposed to thousands of color natural scenes from the extensively annotated Microsoft Common Objects in Context (COCO)[20] images during 30–40 scan sessions. We focus on data from four subjects who viewed identical stimuli, ensuring consistency in our analysis. This dataset is instrumental for investigating brain visual perception and pattern recognition.

### 4.2.2 Framework

Our framework integrates advanced feature extraction and dimension reduction techniques to analyze complex visual stimuli. We use pre-trained models, ResNet-50[16] and DINOv2[30], to extract visual features of a scene, followed by PCA[30] and UMAP[25], respectively, to reduce the dimension to two. For decoding analysis, we use the Nilearn[9]’s implementation of SpaceNet Decoder with Graph-Net regularization[13] to create both classification and regression weight maps. This methodology aids in producing interpretable brain weight maps. The overall pipeline is illustrated in Figure 4.1.

### 4.2.3 Post-hoc classification test

To quantify the informativeness of the resulting weight map from both methods, we use a post-hoc test to evaluate how well class-related information is preserved in the weight maps obtained from decoding analyses. Once we combine a final weight map from a decoding analysis by taking the max magnitude across all the sub-weight

maps, regardless of the decoding target, we use it as a selection mask and decode the scene class from the fMRI analysis again. If the class information is preserved well in the first decoding step, the second post-hoc classification evaluation will yield high prediction accuracy. We evaluate at a number of different sparsity levels by thresholding the resulting weight maps at different levels.

## 4.3 Results

### 4.3.1 Post-hoc classification test

In our post-hoc classification evaluation of the weight maps, our label-free brain decoding method produces very similar levels of F1 score compared to traditional classification-based decoding, shown in Figure 4.2.b. Note that for a fair comparison, we selected an equivalent number of voxels from both the regression-based and classification-based methods. This indicates that the class-related information is adequately preserved in our method, even though this is not explicitly optimized for classification.

### 4.3.2 Analysis of the Visual Cortex Regions

In general, our method produces a similar weight distribution as classification-based decoding. As we can see from figure 4.2.a, we investigate different sub-regions of the visual part, including V1 to V5 cortex according to the Juelich atlas, and the following regions according to the Harvard-Oxford atlas: LG (Lingual Gyrus), LO-1 (Lateral Occipital Cortex superior division), LO-2 (Lateral Occipital Cortex inferior division), IC (Intracalcarine Cortex), CC (Cuneal Cortex), TOF (Temporal

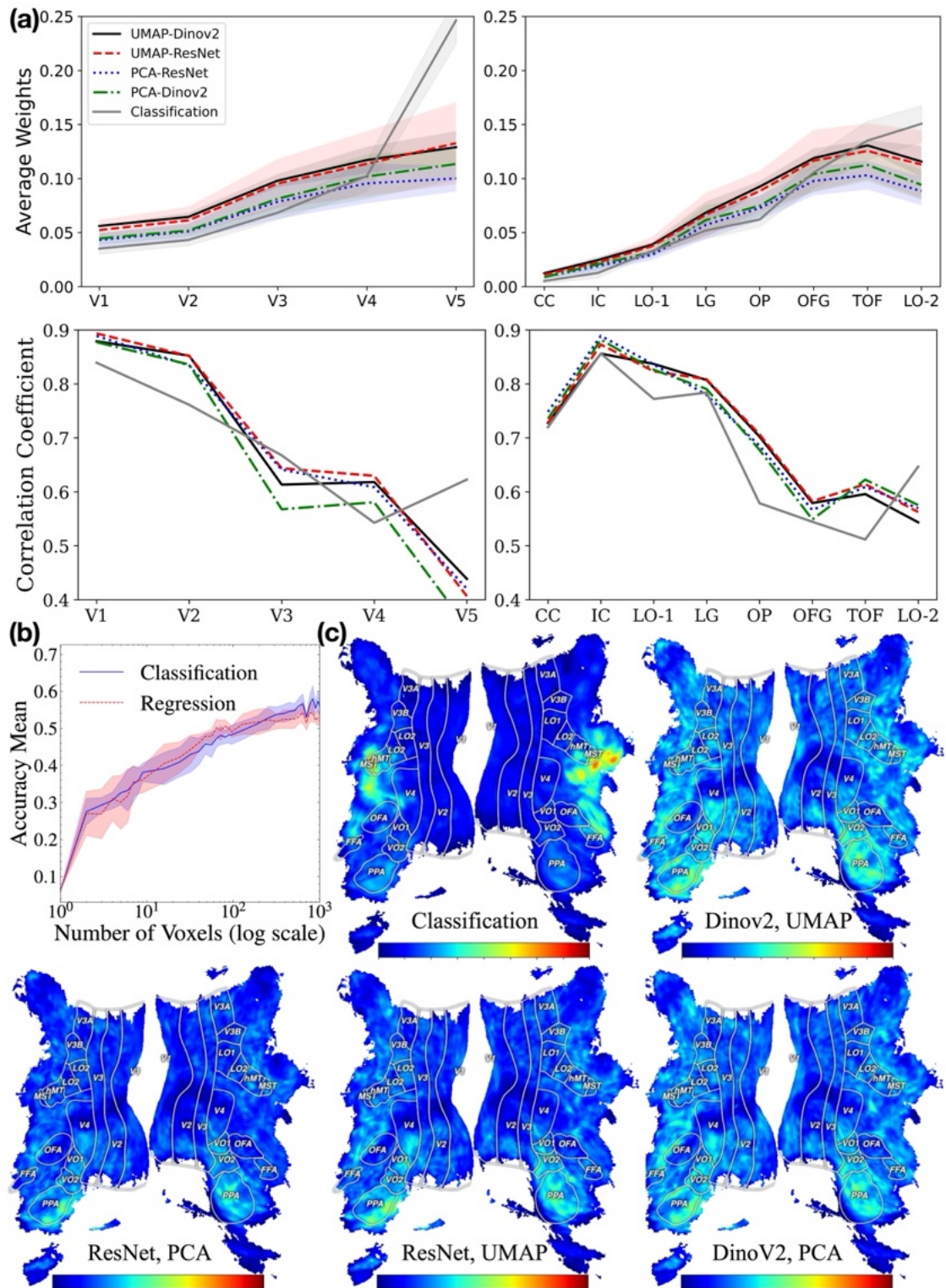


Figure 4.2: (a). Average voxel weights and the mean of weight correlation coefficients across subjects for visual subregions. (b). Image Classification Accuracy (c). Comparative Analysis of Weight Maps Across Methods: Average normalized values from the weight maps of each method across all subjects.

Occipital Fusiform Cortex), OFG (Occipital Fusiform Gyrus), and OP (Occipital Pole). Figure 4.2.a shows a high consistency across all combinations of DNNs and dimensionality reduction methods, with a higher average weight in areas associated with higher-level visual processing. While the findings from the classification-based decoding largely corroborate our results, disagreements appear in the V5 cortex and the LO-2 region, both associated with higher-order visual functions [24, 43].

Figure 4.2.a shows the mean correlation between voxel weight maps of different subjects. It is consistent across different subjects but decreases from 0.9 to 0.5 from V1 to V5. This trend might indicate a divergence in how subjects interpret combinations of high-level visual features but share similar processing of low-level visual features.

The visualization in Figure 4.2.c highlights significant weights in the Parahippocampal Place Area (PPA), a region integral to scene recognition and spatial memory, by the proposed approach. Higher weights are also prominent in the VO1 (visual occipital 1) and VO2 (visual occipital 2) regions, known for their roles in color recognition [37]. Conversely, the classification-based method assigns heavier weights to the Medial Temporal area, emphasizing motion perception[5]. We note that the underlying task of these fMRI scans is scene recollection, where participants recall previously viewed stimuli. This difference suggests that the weight maps produced by our method are better aligned with the underlying task of scene recognition.

Further analysis of weight progression from visual areas V1 to V5 shows an increase in weight intensity from basic visual processing in V1 to complex integrations in V5. This is particularly evident in classification-based methods, notably in the hMT(human Middle Temporal)/MST(Medial Superior Temporal) area known

for motion sensitivity. This pattern highlights different neural engagements based on the decoding strategy, illustrating how these methods process visual information differently. This research enhances our understanding of the visual cortex’s functional architecture and demonstrates the potential of advanced decoding techniques to reflect cognitive tasks in visual processing more accurately.

## 4.4 Conclusion and Discussion

We introduce a novel label-free brain decoding methodology using the Natural Scenes Dataset (NSD), where we replace the commonly used classification targets with features from pre-trained deep neural networks, which removes the need for predefined classes or labels[38]. We demonstrated that this approach yields weight maps as informative as the traditional classification-based methods. A comparison of the weight maps shows that the regression-based method assigns weights in a way that better captures the underlying task of scene recognition, notably in brain regions like the Parahippocampal Place Area (PPA). Our proposed method provides a decoding analysis method that preserves relevant visual information, is consistent across parameter choices, and removes the reliance on hand-designed labels.

## Chapter 5

# Conclusions and Future Work

The explorations and experiments conducted in this thesis provide initial insights into the capacities of neural decoding and representational learning across both auditory and visual modalities. While promising, these findings should be considered preliminary, indicating potential pathways and hypotheses rather than conclusive evidence.

The methodologies introduced across different chapters, from disentangled representations in speech processing with ECoG signals to innovative approaches in visual cortex mapping, highlight the complexity and potential of decoding neural signals. However, it is crucial to recognize the limitations and challenges encountered, such as overfitting in specific models and the need for more expansive and varied datasets to validate and generalize these results.

Given these considerations, our work sets the stage for further research. Extending these studies to include a broader range of subjects and more diverse neural data will be essential to enhance the robustness and applicability of the proposed models. Additionally, integrating more sophisticated machine learning



techniques and exploring new neural network architectures could address some of the shortcomings identified in our initial experiments.

Future work will also need to delve deeper into the mechanisms underlying the disentangled representations and their implications for practical applications in neural engineering, such as neuroprosthetics and brain-computer interfaces. By continuing to refine the techniques and expand the scope of our investigations, we aim to contribute more definitively to the field, paving the way for more effective and nuanced technologies in both medical diagnostics and interactive computing systems.

# Bibliography

- [1] The year of brain–computer interfaces. *Nature Electronics*, 6:643, 2023.
- [2] E. J. Allen, G. St-Yves, Y. Wu, J. L. Breedlove, J. S. Prince, L. T. Dowdle, M. Nau, B. Caron, F. Pestilli, I. Charest, J. B. Hutchinson, T. Naselaris, and K. Kay. A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience*, 25(1):116–126, Jan. 2022.
- [3] A. Bardes, J. Ponce, and Y. LeCun. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning, Jan. 2022. arXiv:2105.04906 [cs].
- [4] K. Bhadra, A. L. Giraud, and S. Marchesotti. Learning to operate an imagined speech brain-computer interface involves the spatial and frequency tuning of neural activity. *bioRxiv*, 2023.
- [5] J. Blumberg and G. Kreiman. How cortical neurons help us see: visual recognition in the human brain. *The Journal of Clinical Investigation*, 120(9):3054–3063, 9 2010.
- [6] T. A. Carlson, P. Schrater, and S. He. Patterns of activity in the categorical representations of objects. *Journal of cognitive neuroscience*, 15(5):704–717, 2003.

- [7] J. Chen, X. Chen, R. Wang, C. Le, A. Khalilian-Gourtani, E. Jensen, P. Dugan, W. Doyle, O. Devinsky, D. Friedman, A. Flinker, and Y. Wang. Subject-agnostic transformer-based neural speech decoding from surface and depth electrode signals, 03 2024.
- [8] X. Chen, R. Wang, A. Khalilian-Gourtani, L. Yu, P. Dugan, D. Friedman, W. Doyle, O. Devinsky, Y. Wang, and A. Flinker. A Neural Speech Decoding Framework Leveraging Deep Learning and Speech Synthesis, Sept. 2023. Pages: 2023.09.16.558028 Section: New Results.
- [9] N. contributors. nilearn.
- [10] D. D. Cox and R. L. Savoy. Functional magnetic resonance imaging (fmri) “brain reading”: detecting and classifying distributed patterns of fmri activity in human visual cortex. *Neuroimage*, 19(2):261–270, 2003.
- [11] B. Desplanques, J. Thienpondt, and K. Demuynck. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. In *Interspeech 2020*, interspeech 2020. ISCA, Oct. 2020.
- [12] Y. Ge, S. Abu-El-Haija, G. Xin, and L. Itti. Zero-shot Synthesis with Group-Supervised Learning, Feb. 2021. arXiv:2009.06586 [cs].
- [13] L. Grosenick, B. Klingenberg, K. Katovich, B. Knutson, and J. E. Taylor. Interpretable whole-brain prediction analysis with GraphNet. *NeuroImage*, 72:304–321, May 2013.
- [14] J. V. Haxby, M. I. Gobbini, M. L. Furey, A. Ishai, J. L. Schouten, and P. Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293(5539):2425–2430, 2001.

- [15] J.-D. Haynes and G. Rees. Predicting the orientation of invisible stimuli from activity in human primary visual cortex. *Nature neuroscience*, 8(5):686–691, 2005.
- [16] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition, Dec. 2015. arXiv:1512.03385 [cs].
- [17] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units, June 2021. arXiv:2106.07447 [cs, eess].
- [18] Y. Kamitani and F. Tong. Decoding the visual and subjective contents of the human brain. *Nature neuroscience*, 8(5):679–685, 2005.
- [19] K. Kasi and S. A. Zahorian. Yet another algorithm for pitch tracking. *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1:I–361–I–364, 2002.
- [20] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft COCO: Common Objects in Context, Feb. 2015. arXiv:1405.0312 [cs].
- [21] R. Liu, M. Azabou, M. Dabagia, C.-H. Lin, M. G. Azar, K. B. Hengen, M. Valko, and E. L. Dyer. Drop, Swap, and Generate: A Self-Supervised Approach for Generating Neural Activity, Nov. 2021. arXiv:2111.02338 [cs].
- [22] Y. Liu, Y. Ma, W. Zhou, G. Zhu, and N. Zheng. Brainclip: Bridging brain and visual-linguistic representation via clip for generic natural visual stimulus decoding. arxiv 2023. *arXiv preprint arXiv:2302.12971*.

- [23] L. v. d. Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.
- [24] R. Malach, J. B. Reppas, R. R. Benson, K. K. Kwong, H. Jiang, W. A. Kennedy, P. J. Ledden, T. J. Brady, B. R. Rosen, and R. B. Tootell. Object-related activity revealed by functional magnetic resonance imaging in human occipital cortex. *Proceedings of the National Academy of Sciences*, 92(18):8135–8139, 1995.
- [25] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, Sept. 2020. arXiv:1802.03426 [cs, stat].
- [26] F. Mentzer, D. Minnen, E. Agustsson, and M. Tschannen. Finite scalar quantization: Vq-vae made simple, 2023.
- [27] S. L. Metzger, K. T. Littlejohn, A. B. Silva, D. A. Moses, M. P. Seaton, R. Wang, M. E. Dougherty, J. R. Liu, P. Wu, M. A. Berger, I. Zhuravleva, A. Tu-Chan, K. Ganguly, G. K. Anumanchipalli, and E. F. Chang. A high-performance neuroprosthesis for speech decoding and avatar control. *Nature*, 620(7976):1037–1046, Aug. 2023. Number: 7976 Publisher: Nature Publishing Group.
- [28] M. Mozafari, L. Reddy, and R. VanRullen. Reconstructing natural scenes from fmri patterns using bigbigan. In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2020.
- [29] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *INTERSPEECH*, pages 2616–2620. ISCA, 2017.

- [30] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. DINOv2: Learning Robust Visual Features without Supervision, Feb. 2024. arXiv:2304.07193 [cs].
- [31] F. Ozcelik, B. Choksi, M. Mozafari, L. Reddy, and R. VanRullen. Reconstruction of perceived images from fmri patterns and semantic brain exploration using instance-conditioned gans. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.
- [32] F. Ozcelik and R. VanRullen. Natural scene reconstruction from fmri signals using generative latent diffusion. *Scientific Reports*, 13(1):15666, 2023.
- [33] A. Polyak, Y. Adi, J. Copet, E. Kharitonov, K. Lakhota, W.-N. Hsu, A. Mohamed, and E. Dupoux. Speech Resynthesis from Discrete Disentangled Self-Supervised Representations, July 2021. arXiv:2104.00355 [cs, eess].
- [34] P. S. Scotti, A. Banerjee, J. Goode, S. Shabalin, A. Nguyen, E. Cohen, A. J. Dempster, N. Verlinde, E. Yundler, D. Weisberg, K. A. Norman, and T. M. Abraham. Reconstructing the Mind’s Eye: fMRI-to-Image with Contrastive Learning and Diffusion Priors, Oct. 2023. arXiv:2305.18274 [cs, q-bio].
- [35] G. Shen, T. Horikawa, K. Majima, and Y. Kamitani. Deep image reconstruction from human brain activity. *PLoS computational biology*, 15(1):e1006633, 2019.
- [36] Y. Takagi and S. Nishimoto. High-resolution image reconstruction with latent diffusion models from human brain activity. In *Proceedings of the IEEE/CVF*

*Conference on Computer Vision and Pattern Recognition*, pages 14453–14463, 2023.

- [37] S. N. Tomson, M. Narayan, G. I. Allen, and D. M. Eagleman. Neural networks of colored sequence synesthesia. *Journal of Neuroscience*, 33(35):14098–14106, 2013.
- [38] H. Wen, J. Shi, Y. Zhang, K.-H. Lu, J. Cao, and Z. Liu. Neural Encoding and Decoding with Deep Learning for Dynamic Natural Vision. *Cerebral Cortex*, 28(12):4136–4160, 10 2017.
- [39] F. R. Willett, E. M. Kunz, C. Fan, D. T. Avansino, G. H. Wilson, E. Y. Choi, F. Kamdar, M. F. Glasser, L. R. Hochberg, S. Druckmann, K. V. Shenoy, and J. M. Henderson. A high-performance speech neuroprosthesis. *Nature*, 620(7976):1031–1036, Aug. 2023. Number: 7976 Publisher: Nature Publishing Group.
- [40] J. Yamagishi, C. Veaux, and K. MacDonald. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92). [sound], 2019.
- [41] H. Yang, J. Gee, and J. Shi. Brain Decodes Deep Nets, Mar. 2024. arXiv:2312.01280 [cs].
- [42] H. Yang, J. Gee, and J. Shi. Brain Decodes Deep Nets, Mar. 2024. arXiv:2312.01280 [cs].
- [43] S. Zeki. Functional organization of a visual area in the posterior bank of the superior temporal sulcus of the rhesus monkey. *Journal of Physiology*, 236(3):549–573, 1974.

- [44] M. Śliwowski, M. Martin, A. Souloumiac, P. Blanchart, and T. Aksenova. Impact of dataset size and long-term ecog-based bci usage on deep learning decoders performance. *Frontiers in Human Neuroscience*, 17:1111645, 2023.