

# End-to-End Deep Learning for Structural Brain Imaging: A Unified Framework

Yao Su<sup>1</sup>, Keqi Han<sup>2</sup>, Mingjie Zeng<sup>1</sup>, Lichao Sun<sup>3</sup>, Liang Zhan<sup>4</sup>, Carl Yang<sup>2</sup>, Lifang He<sup>3</sup>, Xiangnan Kong<sup>1</sup>

<sup>1</sup>Worcester Polytechnic Institute, Worcester, MA, USA

<sup>2</sup>Emory University, Atlanta, GA, USA

<sup>3</sup>Lehigh University, Bethlehem, PA, USA

<sup>4</sup>University of Pittsburgh, Pittsburgh, PA, USA

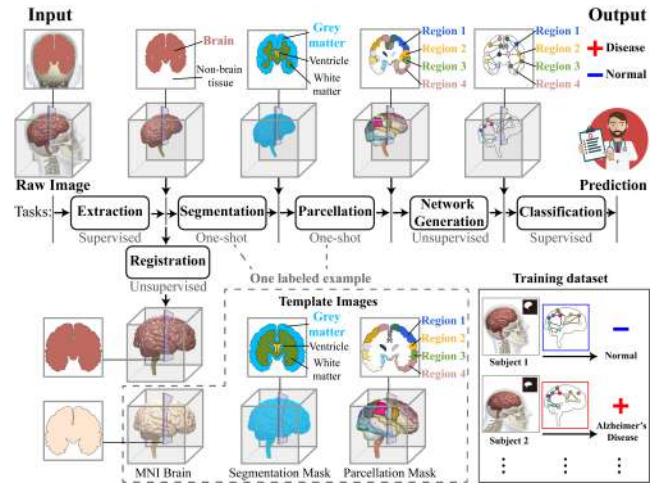
ysu6@wpi.edu, keqi.han@emory.edu, mzeng2@wpi.edu, lis221@lehigh.edu, liang.zhan@pitt.edu, j.carlyang@emory.edu, lih319@lehigh.edu, xkong@wpi.edu

## Abstract

Brain imaging analysis is fundamental in neuroscience, providing valuable insights into brain structure and function. Traditional workflows follow a sequential pipeline—brain extraction, registration, segmentation, parcellation, network generation, and classification—treating each step as an independent task. These methods rely heavily on task-specific training data and expert intervention to correct intermediate errors, making them particularly burdensome for high-dimensional neuroimaging data, where annotations and quality control are costly and time-consuming. We introduce UniBrain, a unified end-to-end framework that integrates all processing steps into a single optimization process, allowing tasks to interact and refine each other. Unlike traditional approaches that require extensive task-specific annotations, UniBrain operates with minimal supervision, leveraging only low-cost labels (*i.e.*, classification and extraction) and a single labeled atlas. By jointly optimizing extraction, registration, segmentation, parcellation, network generation, and classification, UniBrain enhances both accuracy and computational efficiency while significantly reducing annotation effort. Experimental results demonstrate its superiority over existing methods across multiple tasks, offering a more scalable and reliable solution for neuroimaging analysis.

## Introduction

The human brain, with its billions of interconnected neurons that form the connectome, is the foundation of our cognitive functions and behaviors. Understanding this intricate connectivity is crucial for decoding the brain’s mechanisms in development and degeneration. However, accurately mapping the connectome remains a significant challenge due to limitations in current methods. Traditional workflows rely on structural or functional neuroimaging data processed through fragmented steps—brain extraction, registration, segmentation/parcellation, and network generation—often requiring manual quality control, which is costly and represents a critical barrier for quantitative brain biomarkers to enter clinical practice. Furthermore, piecemeal approaches prevent simultaneous optimization of interdependent stages, leading to inefficiencies and limiting the discovery of nuanced connections. Errors introduced in earlier steps propagate through subsequent analyses, resulting in potentially



**Figure 1:** The problem of end-to-end learning for brain imaging tasks. Given a set of raw images, each with a corresponding extraction mask and diagnosis label, along with a labeled template brain (with segmentation and parcellation masks), the goal is to train a model to simultaneously perform extraction, registration, segmentation, parcellation, network generation, and classification tasks.

misleading interpretations of brain dynamics. Moreover, the time-intensive nature of these workflows hinders scalability and efficiency.

Instead of tedious, step-by-step processing for brain imaging data, recent studies support transforming these pipelines into deep neural networks for joint learning and end-to-end optimization (Ren et al. 2024; Agarwal et al. 2022). While several approaches have been proposed—such as joint extraction and registration (Su et al. 2022b), joint registration and parcellation (Zhao et al. 2021; Lord et al. 2007), and joint network generation and disease prediction (Campbell et al. 2022; Mahmood et al. 2021; Kan et al. 2022a)—there is currently no framework that unifies and simultaneously optimizes all these processing stages to directly create brain networks from raw imaging data. Mapping the connectome of human brain as a brain network (*i.e.*, graph), has become one of the most pervasive paradigms in neuroscience (Sporns, Tonomi, and Kotter 2005; Bargmann and Marder 2013). Representing the brain as a graph of nodes (regions) and edges (structural or functional connections) en-

ables gaining critical insights into brain organization, identifying key regions or hubs, and understanding how brain connectivity changes under different conditions (e.g., during development, aging, or neurological disorders) (Kaiser 2011; Crossley et al. 2014; Xu et al. 2015). This need has intensified with the rapidly advancing imaging technologies and massive data collection.

In this paper, we propose UniBrain, the first end-to-end deep learning model that seamlessly integrates brain extraction, registration, segmentation, parcellation, network generation, and clinical classification into a unified optimization process, as illustrated in Figure 1. Our objective is to investigate the interdependence of these tasks, enabling them to enhance each other’s performance while relying on minimal labeled data. Specifically, we leverage low-cost labels (*i.e.*, extraction mask, classification label) and a single labeled template (*a.k.a.* atlas) to jointly optimize all tasks. Notably, our approach eliminates the need for instance-level ground-truth labels for registration, segmentation, parcellation, and network connectivity during model training. Extensive experiments on the public ADHD dataset with 3D brain sMRI demonstrate that our method outperforms state-of-the-art approaches across all six tasks.

## Related Works

In the literature, related tasks in brain imaging analysis have been extensively studied. Conventional methods primarily focus on designing methods for brain extraction (Kleesiek et al. 2016; Lucena et al. 2019), registration (Sokooti et al. 2017; Su et al. 2022a), segmentation (Akkus et al. 2017; Kamnitsas et al. 2017; Chen et al. 2018), parcellation (Thyreau and Taki 2020; Lim et al. 2022), network generation (Škoch et al. 2022; Yin et al. 2023) and classification (Li et al. 2021; Kawahara et al. 2017; Kan et al. 2022b) separately under supervised settings. However, in brain imaging studies, the collection of voxel-level annotations, transformations between images, and task-specific brain networks often prove to be expensive, as it demands extensive expertise, effort, and time to produce accurate labels, especially for high-dimensional neuroimaging data, *e.g.*, 3D MRI. To reduce this high demand for annotations, recent works have utilized automatic extraction tools (Smith 2002; Cox 1996; Shattuck and Leahy 2002; Ségonne et al. 2004), unsupervised registration models (Balakrishnan et al. 2018; Su et al. 2022a), inverse warping (Jaderberg et al. 2015), and correlation-based metrics (Liang et al. 2012) for performing extraction, registration, segmentation, parcellation and network generation. Nevertheless, these pipeline-based approaches frequently rely on manual quality control to correct intermediate results before performing subsequent tasks. Conducting such visual inspections is not only time-consuming and labor-intensive but also suffers from intra- and inter-rater variability, thereby impeding the overall efficiency and performance. More recently, joint extraction and registration (Su et al. 2022b), joint registration and segmentation (Xu and Niethammer 2019), joint extraction, registration and segmentation (Su et al. 2023), and joint network generation and classification (Kan et al. 2022a) have been developed for collective learning. How-

ever, partial joint learning overlooks the potential interrelationships among these tasks, which can adversely affect overall performance and limit generalizability. There is a pressing need for more integrated, automated and robust methodologies that can seamlessly integrate and optimize all stages of raw brain imaging-to-graph analysis within a unified framework.

## Our Approach

UniBrain integrates multiple modules for brain extraction, registration, segmentation, parcellation, network generation, and classification, seamlessly connecting them within an end-to-end framework to enable collective learning. Below, we provide a detailed description of each module.

### Extraction Module

The extraction module aims to extract brain from the raw image with assistance from two components:

**Extraction Network:**  $f_e$ . The extraction network  $f_e(\cdot)$  acts as an annotator, intended to identify brain and non-brain tissues in the source image  $\mathbf{S}$  and delineate their locations, thus providing the guidance for subsequent non-brain tissue elimination. Specifically, we employ the 3D U-Net as the base network to learn  $f_e(\cdot)$ . The process can be formally expressed as:

$$\hat{\mathbf{M}} = f_e(\mathbf{S}), \quad (1)$$

where  $\hat{\mathbf{M}}$  is predicted extraction mask. During inference,  $\hat{\mathbf{M}}$  is binarized by a Heaviside step function.

**Overlay Layer:**  $OL$ . The overlay layer serves to eliminate non-brain tissues by applying the predicted brain mask  $\hat{\mathbf{M}}$  to the source image  $\mathbf{S}$ . The final extracted image is  $\mathbf{E} = \mathbf{S} \circ \hat{\mathbf{M}}$ , where  $\circ$  denotes the element-wise multiplication.

### Registration Module

The registration module aims to align the extracted image with the target image, providing transformations for subsequent segmentation and parcellation tasks. This module comprises two main components:

**Registration Network:**  $f_r$ . The registration network  $f_r(\cdot, \cdot)$  processes the extracted image  $\mathbf{E}$  and target image  $\mathbf{T}$  to learn the affine transformation  $\mathbf{A}$ , which establishes the coordinate correspondence between source and target image space. A 3D CNN-based encoder is used to learn  $f_r(\cdot, \cdot)$  as:

$$\mathbf{A} = f_r(\mathbf{E}, \mathbf{T}). \quad (2)$$

We leverage the multi-stage registration technique (Su et al. 2022a; Zhao et al. 2019) to boost registration performance, where  $\mathbf{E}$  is recursively aligned with  $\mathbf{T}$  through  $M$  stages.

**Spatial Transformation Layer:**  $STL$ . A key step in image registration is reconstructing the warped image  $\mathbf{W}$  from the extracted image  $\mathbf{E}$  using the affine transformation  $\mathbf{A}$ . This warping process is facilitated by a spatial transformation layer (STL), which resamples voxels from the extracted image  $\mathbf{E}$  to produce the warped image  $\mathbf{W} = \mathcal{T}(\mathbf{E}, \mathbf{A})$ . Given the affine transformation operator, we hold

$$\mathbf{W}_{xyz} = \mathbf{E}_{x'y'z'}, \quad (3)$$

where coordinate correspondence  $[x', y', z', 1]^\top = \mathbf{A}[x, y, z, 1]^\top$ . To enable successful gradient propagation, we use a differentiable transformation based on trilinear interpolation proposed by (Jaderberg et al. 2015).

## Segmentation & Parcellation Module

The segmentation and parcellation module creates segmentation and parcellation masks on the source image. Leveraging recent developments in one-shot learning (Wang et al. 2020; Ding, Yu, and Yang 2021; Su et al. 2023), the module can generate these masks using a single labeled template image. The module contains two main components:

**Inverse Warping** Utilizing a single labeled example (*i.e.*, target image  $\mathbf{T}$  with its corresponding segmentation mask  $\mathbf{B}$  and parcellation mask  $\mathbf{P}$ ) and the learned affine transformation  $\mathbf{A}$ , we apply the inverse transformation  $\mathbf{A}^{-1}$  to generate warped segmentation mask  $\mathbf{V} = \mathcal{T}(\mathbf{B}, \mathbf{A}^{-1})$  and parcellation mask  $\mathbf{U} = \mathcal{T}(\mathbf{P}, \mathbf{A}^{-1})$  in the source image space as:

$$\mathbf{V}_{cxyz} = \mathbf{B}_{cx'y'z'}, \forall c \in \{1, \dots, C\}, \quad (4)$$

$$\mathbf{U}_{kxyz} = \mathbf{P}_{kx'y'z'}, \forall k \in \{1, \dots, K\}, \quad (5)$$

where coordinate correspondence  $[x', y', z', 1]^\top = \mathbf{A}^{-1}[x, y, z, 1]^\top$ ,  $c$  is the index for tissue class and  $k$  is the index for ROIs. Same as the STL layer in Registration Module, we then apply a differentiable transformation based on trilinear interpolation.

**Segmentation Network:**  $f_s$ . The segmentation network  $f_s(\cdot)$  aims to generate a segmentation mask for the source image  $\mathbf{S}$  that matches the synthesized warped segmentation mask  $\mathbf{V}$ . We employ the widely-used 3D U-Net as the base network to learn  $f_s(\cdot)$ . Formally, we have:

$$\mathbf{R} = f_s(\mathbf{S}). \quad (6)$$

## Brain Network Module

The brain network module generates the brain network using ROI information from parcellation mask  $\mathbf{U}$  and the source image  $\mathbf{S}$ . The modules include three components:

**Overlay Layer:**  $OL$ . Similar to  $OL$  in the Extraction Module, this component is responsible for isolating each ROI from the source image  $\mathbf{S}$  using parcellation mask  $\mathbf{U}$ . The parcellated image  $\mathbf{F} = \mathbf{S} \circ \mathbf{U}$  is generated by applying an element-wise product  $\circ$  between  $\mathbf{S}$  and  $\mathbf{U}$ .

**Brain Network Function:**  $f_o$ . The brain network function aims to learn the representation for each ROI within the parcellated image  $\mathbf{F}$ . A weight-sharing Multilayer Perceptron (MLP) is employed to learn  $f_o(\cdot)$ , ensuring consistent feature extraction and generalization, which is expressed as:

$$\mathbf{H}_k = f_o(\mathbf{F}_k), \forall k \in \{1, \dots, K\}, \quad (7)$$

where  $k$  is the index for the ROIs.

**Brain Network Generation.** The step generates a brain network based on the similarity between ROI representation pairs. Without loss of generality, here we use inner-product to measure the edge weights of the brain network. However, other differentiable similarity functions (*e.g.*, Mahalanobis distance and cosine similarity) can be used. To compute the

connectivity matrix  $\mathbf{C}$ , each ROI representation  $\mathbf{H}_k$  is first normalized with the  $\ell^2$ -norm, followed by the inner-product:

$$\mathbf{C} = \mathbf{H}\mathbf{H}^\top. \quad (8)$$

This normalization scales the values of  $\mathbf{C}$  to the range of  $[-1, 1]$ , ensuring the stabilization of the learning process and maintaining consistent weight magnitudes in the network.

## Classification Module

The classification module makes a final predictive diagnosis.

**Classification Network:**  $f_g$ . The classification network  $f_g(\cdot, \cdot)$  aims to make a prediction based on the generated brain network while feeding task-specific insights to the preceding module, facilitating the brain network generation. We leverage the GCN (Kipf and Welling 2017) as the base network. The prediction  $\hat{y}$  is obtained as:

$$\hat{y} = f_g(\mathbf{C}, \mathbf{H}), \quad (9)$$

where  $\mathbf{H}$  is the initial node features and  $\mathbf{C}$  is the learnable connectivity matrix provided by the brain network module.

## End-to-End Training

We train UniBrain by minimizing the objective function as:

$$\mathcal{L} = \mathcal{L}_{cls}(\hat{y}, y) + \alpha \mathcal{L}_{ext}(\hat{\mathbf{M}}, \mathbf{M}) + \beta \mathcal{L}_{sim}(\mathbf{W}, \mathbf{T}) + \gamma \mathcal{L}_{seg}(\mathbf{R}, \mathbf{V}), \quad (10)$$

where  $\mathcal{L}_{cls}(\cdot, \cdot)$  is classification loss term,  $\mathcal{L}_{ext}(\cdot, \cdot)$  is extraction loss term,  $\mathcal{L}_{sim}(\cdot, \cdot)$  is image dissimilarity loss term, and  $\mathcal{L}_{seg}(\cdot, \cdot)$  is segmentation loss term. This equation incorporates bidirectional supervision ( $\mathcal{L}_{cls}(\cdot, \cdot)$  and  $\mathcal{L}_{ext}(\cdot, \cdot)$ ), which envelops the entire network to ensure positive forward propagation and controllable feedback across tasks. Additionally, unsupervised and one-shot guidance ( $\mathcal{L}_{sim}(\cdot, \cdot)$  and  $\mathcal{L}_{seg}(\cdot, \cdot)$ ) within the model reduces reliance on high-cost annotations. The loss terms are scaled by  $\alpha$ ,  $\beta$ , and  $\gamma$  to balance their impacts.

By leveraging the differentiability in each component of this design, our model achieves joint optimization in an end-to-end manner. All tasks are unified within a single model for collective learning, mutually boosting their performance with limited labels.

## Experiments

### Experimental Settings

**Datasets.** We evaluate the effectiveness of our proposed method on the public real-world ADHD dataset with 3D brain sMRI (consortium 2012). The dataset contains records for 776 subjects, labeled as real patients (positive) and normal controls (negative). The original dataset is unbalanced, following (Kong et al. 2013), we randomly sampled 100 ADHD patients and 100 normal controls from the dataset for performance evaluation. Out of the 200 scans, 160 are used for training, 20 for validation, and 20 for testing. All scans are cropped and resized to  $96 \times 96 \times 96$  dimensions. We use MNI 152 with the AAL atlas (Tzourio-Mazoyer et al. 2002) as the template image for registration and parcellation.

**Table 1:** Results on ADHD dataset. The results are reported as (mean  $\pm$  std) of each task for each compared method. “+” indicates combining different baselines for the corresponding tasks.

Methods	Extraction		Registration		Segmentation		Parcellation		Classification	
	Dice $\uparrow$	Jaccard $\uparrow$	MI $\uparrow$	CC $\uparrow$	Dice $\uparrow$	Jaccard $\uparrow$	Dice $\uparrow$	Jaccard $\uparrow$	ACC $\uparrow$	AUC-ROC $\uparrow$
BET + FLIRT + DW + KNN + GCN	0.830 $\pm$ 0.058	0.713 $\pm$ 0.079	0.585 $\pm$ 0.031	0.882 $\pm$ 0.041	0.431 $\pm$ 0.058	0.293 $\pm$ 0.049	0.510 $\pm$ 0.172	0.375 $\pm$ 0.142	0.582 $\pm$ 0.034	0.546 $\pm$ 0.028
Synth + FLIRT + DW + KNN + GCN	0.920 $\pm$ 0.012	0.853 $\pm$ 0.021	0.621 $\pm$ 0.018	0.942 $\pm$ 0.006	0.494 $\pm$ 0.015	0.347 $\pm$ 0.013	0.678 $\pm$ 0.040	0.525 $\pm$ 0.040	0.595 $\pm$ 0.043	0.612 $\pm$ 0.024
BET + VM + DW + KNN + GCN	0.830 $\pm$ 0.058	0.713 $\pm$ 0.079	0.584 $\pm$ 0.037	0.874 $\pm$ 0.043	0.432 $\pm$ 0.029	0.296 $\pm$ 0.026	0.599 $\pm$ 0.070	0.442 $\pm$ 0.066	0.578 $\pm$ 0.027	0.568 $\pm$ 0.016
Synth + VM + DW + KNN + GCN	0.920 $\pm$ 0.012	0.853 $\pm$ 0.021	0.632 $\pm$ 0.020	0.940 $\pm$ 0.007	0.447 $\pm$ 0.014	0.309 $\pm$ 0.013	0.619 $\pm$ 0.044	0.463 $\pm$ 0.039	0.582 $\pm$ 0.055	0.598 $\pm$ 0.015
BET + ABN + DW + KNN + GCN	0.830 $\pm$ 0.058	0.713 $\pm$ 0.079	0.585 $\pm$ 0.036	0.877 $\pm$ 0.043	0.446 $\pm$ 0.031	0.308 $\pm$ 0.027	0.653 $\pm$ 0.051	0.497 $\pm$ 0.051	0.526 $\pm$ 0.036	0.571 $\pm$ 0.017
Synth + ABN + DW + KNN + GCN	0.920 $\pm$ 0.012	0.853 $\pm$ 0.021	0.635 $\pm$ 0.021	0.943 $\pm$ 0.009	0.455 $\pm$ 0.015	0.317 $\pm$ 0.013	0.675 $\pm$ 0.026	0.521 $\pm$ 0.027	0.595 $\pm$ 0.039	0.612 $\pm$ 0.012
ERNet + DW + KNN + GCN	0.935 $\pm$ 0.016	0.879 $\pm$ 0.028	0.636 $\pm$ 0.014	0.952 $\pm$ 0.009	0.498 $\pm$ 0.014	0.350 $\pm$ 0.014	0.677 $\pm$ 0.045	0.523 $\pm$ 0.047	0.582 $\pm$ 0.070	0.612 $\pm$ 0.015
BET + DeepAtlas + DW + KNN + GCN	0.830 $\pm$ 0.058	0.713 $\pm$ 0.079	0.587 $\pm$ 0.037	0.874 $\pm$ 0.041	0.478 $\pm$ 0.029	0.344 $\pm$ 0.028	0.591 $\pm$ 0.069	0.434 $\pm$ 0.065	0.599 $\pm$ 0.017	0.579 $\pm$ 0.013
Synth + DeepAtlas + DW + KNN + GCN	0.920 $\pm$ 0.012	0.853 $\pm$ 0.021	0.632 $\pm$ 0.021	0.940 $\pm$ 0.007	0.480 $\pm$ 0.016	0.348 $\pm$ 0.015	0.654 $\pm$ 0.030	0.497 $\pm$ 0.031	0.621 $\pm$ 0.047	0.647 $\pm$ 0.012
JERS + DW + KNN + GCN	0.938 $\pm$ 0.014	0.883 $\pm$ 0.025	0.637 $\pm$ 0.014	0.952 $\pm$ 0.009	0.504 $\pm$ 0.013	0.369 $\pm$ 0.013	0.681 $\pm$ 0.043	0.527 $\pm$ 0.045	0.626 $\pm$ 0.039	0.584 $\pm$ 0.009
JERS + DW + KNN + BGN	0.938 $\pm$ 0.014	0.883 $\pm$ 0.025	0.637 $\pm$ 0.014	0.952 $\pm$ 0.009	0.504 $\pm$ 0.013	0.369 $\pm$ 0.013	0.681 $\pm$ 0.043	0.527 $\pm$ 0.045	0.548 $\pm$ 0.085	0.582 $\pm$ 0.094
JERS + DW + KNN + BNT	0.938 $\pm$ 0.014	0.883 $\pm$ 0.025	0.637 $\pm$ 0.014	0.952 $\pm$ 0.009	0.504 $\pm$ 0.013	0.369 $\pm$ 0.013	0.681 $\pm$ 0.043	0.527 $\pm$ 0.045	0.535 $\pm$ 0.039	0.585 $\pm$ 0.034
<b>UniBrain (ours)</b>	<b>0.970 <math>\pm</math> 0.003</b>	<b>0.942 <math>\pm</math> 0.006</b>	<b>0.652 <math>\pm</math> 0.008</b>	<b>0.957 <math>\pm</math> 0.008</b>	<b>0.520 <math>\pm</math> 0.013</b>	<b>0.381 <math>\pm</math> 0.013</b>	<b>0.708 <math>\pm</math> 0.019</b>	<b>0.557 <math>\pm</math> 0.022</b>	<b>0.652 <math>\pm</math> 0.027</b>	<b>0.712 <math>\pm</math> 0.030</b>

**Table 2:** Summary of compared methods.

Methods	Extraction	Registration	Segmentation	Parcellation	Network Generation	Classification
BET	✓				✗	
SynthStrip	✓	✗			✗	
FLIRT	✗	✓			✗	
VM	✗	✓			✗	
ABN	✗	✓			✗	
DW	✗	✗	✓	✓	✗	✗
DeepAtlas	✗	✓	✓	✗	✗	✗
ERNet	✓	✓	✗	✗	✗	✗
JERS	✓	✓	✓	✗	✗	✗
KNN	✗	✗	✗	✗	✓	✗
GCN	✗	✗	✗	✗	✗	✓
BGN	✗	✗	✗	✗	✗	✓
BNT	✗	✗	✗	✗	✗	✓
<b>UniBrain</b>	✓	✓	✓	✓	✓	✓

**Compared Methods.** We compare our UniBrain with several representative baselines. 1) Extraction: BET (Smith 2002) and SynthStrip (Hoopes et al. 2022); 2) Registration: FLIRT (Jenkinson and Smith 2001), VM (Balakrishnan et al. 2018) and ABN (Su et al. 2022a); 3) Segmentation and Parcellation: DW (Jaderberg et al. 2015); 4) Network Generation (Zhou et al. 2022); 5) Classification: GCN (Kipf and Welling 2017), BGN (Li et al. 2021) and BNT (Kan et al. 2022b); 6) Partial Joint: DeepAtlas (Registration-Segmentation) (Xu and Niethammer 2019), ERNet (Extraction-Registration) (Su et al. 2022b) and JERS (Extraction-Segmentation-Segmentation) (Su et al. 2023). Notably, there are no existing solutions that can simultaneously perform all tasks in an end-to-end framework. Thus, for comparison, we designed a pipeline-based solution by combining different state-of-the-art methods for each task. The summary of baselines is shown in Table 2.

**Implementation.** Our experiments are conducted on Ubuntu 20.04 LTS, utilizing an AMD EPYC 7543 CPU and an NVIDIA Tesla A100-80G GPU. We split the datasets into training, validation, and test sets as introduced in the Datasets section. The training set is for learning model parameters, the validation set evaluates hyperparameter settings (e.g., loss term weights), and the test set is used only once to report the final evaluation results. The code is implemented in Python 3.7.6, and the neural networks are built using PyTorch 1.7.1. The source code is available at <https://github.com/Anonymous7852/UniBrain>.

## Experimental Results

We compare UniBrain with baseline methods in terms of extraction, registration, segmentation, parcellation, and classification accuracy and efficiency. Additionally, we evaluate

**Table 3:** Running time of compared methods on ADHD dataset.

Methods	Time (Sec) ↓					
	Ext	Reg	Seg	Parc	NG	
BET + FLIRT + DW + KNN + GCN	1.2	4.1	1.2 $\times$ 10 <sup>-1</sup>	1.5 $\times$ 10 <sup>-1</sup>	8.2 $\times$ 10 <sup>-1</sup>	2.0 $\times$ 10 <sup>-5</sup>
Synth + FLIRT + DW + KNN + GCN	9.8	5.2	1.3 $\times$ 10 <sup>-1</sup>	1.6 $\times$ 10 <sup>-1</sup>	7.9 $\times$ 10 <sup>-1</sup>	3.6 $\times$ 10 <sup>-5</sup>
BET + VM + DW + KNN + GCN	1.2	5.7 $\times$ 10 <sup>-3</sup>	1.0 $\times$ 10 <sup>-4</sup>	1.1 $\times$ 10 <sup>-4</sup>	7.9 $\times$ 10 <sup>-1</sup>	4.0 $\times$ 10 <sup>-5</sup>
Synth + ABN + DW + KNN + GCN	9.8	9.2 $\times$ 10 <sup>-3</sup>	1.0 $\times$ 10 <sup>-4</sup>	1.1 $\times$ 10 <sup>-4</sup>	8.2 $\times$ 10 <sup>-1</sup>	3.4 $\times$ 10 <sup>-5</sup>
ERNet + DW + KNN + GCN	4.0 $\times$ 10 <sup>-2</sup>		1.0 $\times$ 10 <sup>-4</sup>	1.1 $\times$ 10 <sup>-4</sup>	8.0 $\times$ 10 <sup>-1</sup>	3.1 $\times$ 10 <sup>-5</sup>
Synth + DeepAtlas + DW + KNN + GCN	9.8		7.5 $\times$ 10 <sup>-3</sup>	1.1 $\times$ 10 <sup>-4</sup>	8.1 $\times$ 10 <sup>-1</sup>	4.4 $\times$ 10 <sup>-5</sup>
JERS + DW + KNN + GCN		4.9 $\times$ 10 <sup>-2</sup>		1.1 $\times$ 10 <sup>-4</sup>	8.1 $\times$ 10 <sup>-1</sup>	7.4 $\times$ 10 <sup>-5</sup>
JERS + DW + KNN + BGN		4.9 $\times$ 10 <sup>-2</sup>		1.1 $\times$ 10 <sup>-4</sup>	8.1 $\times$ 10 <sup>-1</sup>	2.4 $\times$ 10 <sup>-3</sup>
JERS + DW + KNN + BNT		4.9 $\times$ 10 <sup>-2</sup>		1.1 $\times$ 10 <sup>-4</sup>	8.0 $\times$ 10 <sup>-1</sup>	4.2 $\times$ 10 <sup>-4</sup>
<b>UniBrain (ours)</b>						2.2 $\times$ 10 <sup>-1</sup>

UniBrain against voxel-based end-to-end brain imaging analysis solutions, which bypass brain network generation and rely solely on voxel-level information from images for predictions. Experimental results show that: 1) UniBrain consistently outperforms other methods in extraction, registration, segmentation, parcellation, and classification, while also being time-efficient; 2) UniBrain also surpasses voxel-based end-to-end brain imaging analysis solutions. Similar results were also observed on the ABIDE (Tyszka et al. 2014) datasets. Due to space constraints, we will present these findings in detail in a future journal publication.

**Overall Results.** Table 1 show the results of the compared methods and the proposed UniBrain in extraction, registration, segmentation, parcellation, and classification tasks. Based on the comprehensive evaluation on the public dataset, UniBrain outperforms existing methods in all tasks. 1) In extraction, we observed that joint-based extraction methods (ERNet, JERS and UniBrain) outperform single-stage extraction methods (BET and Synth). Specifically, UniBrain achieves up to a 5.4% improvement in extraction dice scores over the best single-stage method Synth. 2) For the registration task, methods with strong extraction results typically yield better registration accuracy, highlighting the dependency of accurate registration on prior extraction quality. 3) Good registration enhances segmentation and parcellation performance, as these tasks rely on accurate registration. 4) Classification task results also reflect this trend, with higher parcellation accuracy (like Synth-based, JERS-based, UniBrain) yielding better outcomes due to the classification network leveraging parcellation masks for brain network construction. Overall, there’s a clear interdependence among brain imaging analysis tasks, with strengths and errors propagating across them. Partially joint methods like ERNet, JERS, and DeepAtlas show improved performance in their joint tasks but are limited when combined with other separate models. In contrast, UniBrain, benefiting from full end-to-end joint learning, uniquely excels across all tasks.

**Table 4:** Voxel-based End-to-End Learning on ADHD dataset

Methods	Extraction		Registration		Classification	
	Dice $\uparrow$	Jaccard $\uparrow$	MI $\uparrow$	CC $\uparrow$	ACC $\uparrow$	AUC-ROC $\uparrow$
3D-CNN	-	-	-	-	$0.539 \pm 0.048$	$0.623 \pm 0.014$
BET + 3D-CNN	$0.830 \pm 0.058$	$0.713 \pm 0.079$	-	-	$0.539 \pm 0.021$	$0.587 \pm 0.019$
Synth + 3D-CNN	$0.920 \pm 0.012$	$0.853 \pm 0.021$	-	-	$0.547 \pm 0.034$	$0.634 \pm 0.018$
ERNet <sub>ext</sub> + 3D-CNN	$0.935 \pm 0.016$	$0.879 \pm 0.028$	-	-	$0.582 \pm 0.044$	$0.656 \pm 0.025$
JERS <sub>ext</sub> + 3D-CNN	$0.938 \pm 0.014$	$0.883 \pm 0.025$	-	-	$0.573 \pm 0.037$	$0.638 \pm 0.020$
Synth + FLIRT + 3D-CNN	$0.920 \pm 0.012$	$0.853 \pm 0.021$	$0.621 \pm 0.018$	$0.942 \pm 0.006$	$0.647 \pm 0.056$	$0.656 \pm 0.051$
Synth + VM + 3D-CNN	$0.920 \pm 0.012$	$0.853 \pm 0.021$	$0.632 \pm 0.020$	$0.940 \pm 0.007$	$0.617 \pm 0.060$	$0.651 \pm 0.029$
Synth + ABN + 3D-CNN	$0.920 \pm 0.012$	$0.853 \pm 0.021$	$0.635 \pm 0.021$	$0.943 \pm 0.009$	$0.634 \pm 0.028$	$0.622 \pm 0.019$
Synth + DeepAtlas + 3D-CNN	$0.920 \pm 0.012$	$0.853 \pm 0.021$	$0.632 \pm 0.021$	$0.940 \pm 0.007$	$0.645 \pm 0.039$	$0.642 \pm 0.031$
ERNet + 3D-CNN	$0.935 \pm 0.016$	$0.879 \pm 0.028$	$0.636 \pm 0.014$	$0.952 \pm 0.009$	$0.573 \pm 0.042$	$0.570 \pm 0.022$
JERS + 3D-CNN	$0.938 \pm 0.014$	$0.883 \pm 0.025$	$0.637 \pm 0.014$	$0.952 \pm 0.009$	$0.613 \pm 0.049$	$0.596 \pm 0.025$
<b>UniBrain (ours)</b>	<b><math>0.970 \pm 0.003</math></b>	<b><math>0.942 \pm 0.006</math></b>	<b><math>0.652 \pm 0.008</math></b>	<b><math>0.957 \pm 0.008</math></b>	<b><math>0.652 \pm 0.027</math></b>	<b><math>0.712 \pm 0.030</math></b>

**Running Efficiency.** We measure the efficiency of UniBrain by comparing its inference time with other baselines. The measurement is made on the same device with an AMD EPYC 7543 CPU and an NVIDIA Tesla A100 GPU. The running time is reported as the average processing time for each image in its corresponding task. As indicated in Table 3, fully separate methods are the slowest due to the need for individual optimization of each task. Partially joint learning methods demonstrate increased speed in their joined tasks but still require combination with other methods, limiting overall time efficiency. UniBrain is the fastest method, which efficiently performs all tasks in an end-to-end manner on the same device, enhancing overall speed.

**Voxel-based End-to-End Learning.** We compare UniBrain with voxel-based end-to-end brain imaging analysis solution. In this experiments, we disregard graph-based models, relying only on voxel information from images for final classification predictions. We devised three groups: 1) Direct use of raw MRI images as input (including non-brain tissues, images in different coordinate spaces) for label classification. 2) Use of extracted brain images as input (still in different coordinate spaces) for label classification. 3) Use of the brain been extracted and registered to a standard space as input for classification. As shown in Table 4, we observed that the performance is worse when using raw images as input due to the inclusion of non-brain tissues and spatial transformation noise. Images processed through extraction and registration yielded higher accuracy. UniBrain, integrating preprocessing and classification in a joint learning approach, outperformed all other models.

## Conclusion

This paper presents a novel unified framework, UniBrain, the first end-to-end model to jointly perform a diverse set of brain imaging analysis tasks, including extraction, registration, segmentation, parcellation, network generation and classification. UniBrain integrates heterogeneous information into a single system, enabling efficient knowledge transfer across different modules, and avoiding the need for extensive task-specific labels. Experimental results show that UniBrain outperforms state-of-the-art methods in all tasks while also demonstrating robustness and time efficiency.

## References

Agarwal, D.; Berbis, M. A.; Martín-Noguerol, T.; Luna, A.; Garcia, S. C. P.; and De La Torre-Diez, I. 2022. End-

to-end deep learning architectures using 3D neuroimaging biomarkers for early Alzheimer’s diagnosis. *Mathematics*, 10(15): 2575.

Akkus, Z.; Galimzianova, A.; Hoogi, A.; Rubin, D. L.; and Erickson, B. J. 2017. Deep learning for brain MRI segmentation: state of the art and future directions. *Journal of digital imaging*.

Balakrishnan, G.; Zhao, A.; Sabuncu, M. R.; Guttag, J.; and Dalca, A. V. 2018. An unsupervised learning model for deformable medical image registration. In *CVPR*.

Bargmann, C. I.; and Marder, E. 2013. From the connectome to brain function. *Nature Methods*, 10(6): 483–490.

Campbell, A.; Zippo, A. G.; Passamonti, L.; Toschi, N.; and Lio, P. 2022. Dbgsl: Dynamic brain graph structure learning. In *MDL*.

Chen, H.; Dou, Q.; Yu, L.; Qin, J.; and Heng, P.-A. 2018. VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. *NeuroImage*.

consortium, A.-. 2012. The ADHD-200 consortium: a model to advance the translational potential of neuroimaging in clinical neuroscience. *Frontiers in systems neuroscience*.

Cox, R. W. 1996. AFNI: software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical research*.

Crossley, N. A.; Mechelli, A.; Scott, J.; Carletti, F.; Fox, P. T.; McGuire, P.; and Bullmore, E. T. 2014. The hubs of the human connectome are generally implicated in the anatomy of brain disorders. *Brain*, 137(8): 2382–2395.

Ding, Y.; Yu, X.; and Yang, Y. 2021. Modeling the probabilistic distribution of unlabeled data for one-shot medical image segmentation. In *AAAI*.

Hoopes, A.; Mora, J. S.; Dalca, A. V.; Fischl, B.; and Hoffmann, M. 2022. SynthStrip: Skull-stripping for any brain image. *NeuroImage*.

Jaderberg, M.; Simonyan, K.; Zisserman, A.; and kavukcuoglu, k. 2015. Spatial Transformer Networks. In *NeurIPS*.

Jenkinson, M.; and Smith, S. 2001. A global optimisation method for robust affine registration of brain images. *MIA*.

Kaiser, M. 2011. A tutorial in connectome analysis: topological and spatial features of brain networks. *NeuroImage*, 57(3): 892–907.

Kamnitsas, K.; Ledig, C.; Newcombe, V. F.; Simpson, J. P.; Kane, A. D.; Menon, D. K.; Rueckert, D.; and Glocker, B.

2017. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *MIA*.

Kan, X.; Cui, H.; Lukemire, J.; Guo, Y.; and Yang, C. 2022a. Fbnetgen: Task-aware gnn-based fmri analysis via functional brain network generation. In *MDL*.

Kan, X.; Dai, W.; Cui, H.; Zhang, Z.; Guo, Y.; and Yang, C. 2022b. Brain network transformer. In *NeurIPS*.

Kawahara, J.; Brown, C. J.; Miller, S. P.; Booth, B. G.; Chau, V.; Grunau, R. E.; Zwicker, J. G.; and Hamarneh, G. 2017. BrainNetCNN: Convolutional neural networks for brain networks; towards predicting neurodevelopment. *NeuroImage*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.

Kleesiek, J.; Urban, G.; Hubert, A.; Schwarz, D.; Maier-Hein, K.; Bendszus, M.; and Biller, A. 2016. Deep MRI brain extraction: A 3D convolutional neural network for skull stripping. *NeuroImage*.

Kong, X.; Yu, P. S.; Wang, X.; and Ragin, A. B. 2013. Discriminative feature selection for uncertain graph classification. In *SDM*.

Li, X.; Zhou, Y.; Dvornek, N.; Zhang, M.; Gao, S.; Zhuang, J.; Scheinost, D.; Staib, L. H.; Ventola, P.; and Duncan, J. S. 2021. Braingnn: Interpretable brain graph neural network for fmri analysis. *MIA*.

Liang, X.; Wang, J.; Yan, C.; Shu, N.; Xu, K.; Gong, G.; and He, Y. 2012. Effects of different correlation metrics and preprocessing factors on small-world brain functional networks: a resting-state functional MRI study. *PloS one*.

Lim, E.-C.; Choi, U.-S.; Choi, K. Y.; Lee, J. J.; Sung, Y.-W.; Ogawa, S.; Kim, B. C.; Lee, K. H.; Gim, J.; Initiative, A. D. N.; et al. 2022. DeepParcellation: a novel deep learning method for robust brain magnetic resonance imaging parcellation in older East Asians. *FAN*.

Lord, N. A.; Ho, J.; Vemuri, B. C.; and Eisenschchenk, S. 2007. Simultaneous registration and parcellation of bilateral hippocampal surface pairs for local asymmetry quantification. *TMI*, 26(4): 471–478.

Lucena, O.; Souza, R.; Rittner, L.; Frayne, R.; and Lotufo, R. 2019. Convolutional neural networks for skull-stripping in brain MR imaging using silver standard masks. *AIM*.

Mahmood, U.; Fu, Z.; Calhoun, V. D.; and Plis, S. 2021. A deep learning model for data-driven discovery of functional connectivity. *Algorithms*.

Ren, J.; An, N.; Lin, C.; Zhang, Y.; Sun, Z.; Zhang, W.; Li, S.; Guo, N.; Cui, W.; Hu, Q.; et al. 2024. DeepPrep: An accelerated, scalable, and robust pipeline for neuroimaging preprocessing empowered by deep learning. *bioRxiv*.

Ségonne, F.; Dale, A. M.; Busa, E.; Glessner, M.; Salat, D.; Hahn, H. K.; and Fischl, B. 2004. A hybrid approach to the skull stripping problem in MRI. *NeuroImage*.

Shattuck, D. W.; and Leahy, R. M. 2002. BrainSuite: an automated cortical surface identification tool. *MIA*.

Škoch, A.; Rehák Bučková, B.; Mareš, J.; Tintěra, J.; Sanda, P.; Jajcay, L.; Horáček, J.; Španiel, F.; and Hlinka, J. 2022. Human brain structural connectivity matrices—ready for modelling. *Scientific Data*.

Smith, S. M. 2002. Fast robust automated brain extraction. *Human brain mapping*.

Sokooti, H.; De Vos, B.; Berendsen, F.; Lelieveldt, B. P.; Išgum, I.; and Staring, M. 2017. Nonrigid image registration using multi-scale 3D convolutional neural networks. In *MICCAI*.

Sporns, O.; Tononi, G.; and Kotter, R. 2005. The human connectome: a structural description of the human brain. *PLoS Computational Biology*, 1(4): e42.

Su, Y.; Dai, X.; He, L.; and Kong, X. 2022a. ABN: Anti-Blur Neural Networks for Multi-Stage Deformable Image Registration. In *ICDM*.

Su, Y.; Qian, Z.; He, L.; and Kong, X. 2022b. Ernet: Unsupervised collective extraction and registration in neuroimaging data. In *SIGKDD*.

Su, Y.; Qian, Z.; Ma, L.; He, L.; and Kong, X. 2023. One-shot Joint Extraction, Registration and Segmentation of Neuroimaging Data. In *SIGKDD*.

Thyreau, B.; and Taki, Y. 2020. Learning a cortical parcellation of the brain robust to the MRI segmentation with convolutional neural networks. *MIA*.

Tyszka, J. M.; Kennedy, D. P.; Paul, L. K.; and Adolphs, R. 2014. Largely typical patterns of resting-state functional connectivity in high-functioning adults with autism. *Cerebral cortex*.

Tzourio-Mazoyer, N.; Landeau, B.; Papathanassiou, D.; Crivello, F.; Etard, O.; Delcroix, N.; Mazoyer, B.; and Joliot, M. 2002. Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *NeuroImage*.

Wang, S.; Cao, S.; Wei, D.; Wang, R.; Ma, K.; Wang, L.; Meng, D.; and Zheng, Y. 2020. LT-Net: Label transfer by learning reversible voxel-wise correspondence for one-shot medical image segmentation. In *CVPR*.

Xu, T.; Yang, Z.; Jiang, L.; Xing, X.-X.; and Zuo, X.-N. 2015. A connectome computation system for discovery science of brain. *Science Bulletin*, 60(1): 86–95.

Xu, Z.; and Niethammer, M. 2019. DeepAtlas: Joint semi-supervised learning of image registration and segmentation. In *MICCAI*.

Yin, H.; Su, Y.; Liu, X.; Hartvigsen, T.; Li, Y.; and Kong, X. 2023. Multi-State Brain Network Discovery. In *2023 IEEE International Conference on Big Data (BigData)*, 453–462. IEEE.

Zhao, F.; Wu, Z.; Wang, L.; Lin, W.; Xia, S.; Li, G.; and Consortium, U. B. C. P. 2021. A deep network for joint registration and parcellation of cortical surfaces. In *MICCAI*, 171–181.

Zhao, S.; Dong, Y.; Chang, E. I.; Xu, Y.; et al. 2019. Recursive cascaded networks for unsupervised medical image registration. In *ICCV*.

Zhou, H.; Zhang, Y.; Chen, B. Y.; Shen, L.; and He, L. 2022. Sparse Interpretation of Graph Convolutional Networks for Multi-modal Diagnosis of Alzheimer’s Disease. In *MICCAI*.