Journal of Machine Learning for Biomedical Imaging. 2022:003. pp 1-30 Submitted 09/2021; Published 03/2022
Special Issue: IPMI 2021
Guest Editors: Aasa Feragen, Stefan Sommer, Mads Nielsen, Julia Schnabel

Learning the Effect of Registration Hyperparameters with HyperMorph

Andrew Hoopes ahoopes@mgh.harvard.edu
Martinos Center for Biomedical Imaging, Massachusetts General Hospital

Malte Hoffmann mhoffmann@mgh.harvard.edu
Martinos Center for Biomedical Imaging, Massachusetts General Hospital
Department of Radiology, Harvard Medical School

Douglas N. Greve dgreve@mgh.harvard.edu
Martinos Center for Biomedical Imaging, Massachusetts General Hospital
Department of Radiology, Harvard Medical School

Bruce Fischl bfischl@mgh.harvard.edu
Martinos Center for Biomedical Imaging, Massachusetts General Hospital
Department of Radiology, Harvard Medical School
Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology

John Guttag guttag@mit.edu
Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology

Adrian V. Dalca adalca@mit.edu
Martinos Center for Biomedical Imaging, Massachusetts General Hospital
Department of Radiology, Harvard Medical School
Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology

Abstract

We introduce HyperMorph, a framework that facilitates efficient hyperparameter tuning in learning-based deformable image registration. Classical registration algorithms perform an iterative pair-wise optimization to compute a deformation field that aligns two images. Recent learning-based approaches leverage large image datasets to learn a function that rapidly estimates a deformation for a given image pair. In both strategies, the accuracy of the resulting spatial correspondences is strongly influenced by the choice of certain hyperparameter values. However, an effective hyperparameter search consumes substantial time and human effort as it often involves training multiple models for different fixed hyperparameter values and may lead to suboptimal registration. We propose an amortized hyperparameter learning strategy to alleviate this burden by *learning* the impact of hyperparameters on deformation fields. We design a meta network, or hypernetwork, that predicts the parameters of a registration network for input hyperparameters, thereby comprising a single model that generates the optimal deformation field corresponding to given hyperparameter values. This strategy enables fast, high-resolution hyperparameter search at test-time, reducing the inefficiency of traditional approaches while increasing flexibility. We also demonstrate additional benefits of HyperMorph, including enhanced robustness to model initialization and the ability to rapidly identify optimal hyperparameter values specific to a dataset, image contrast, task, or even anatomical region, all without the need to retrain models. We make our code publicly available at `http://hypermorph.voxelmorph.net`.

Keywords: Hyperparameter Search, Deformable Image Registration, Deep Learning, Weight Sharing, Amortized Learning, Regularization, Hypernetworks

1. Introduction

In deformable image registration, dense correspondences are sought to align two images. Classical iterative registration techniques have been thoroughly studied, leading to mature mathematical frameworks and widely used software packages (Ashburner, 2007; Avants et al., 2008; Beg et al., 2005; Fischl et al., 1999; Rueckert et al., 1999; Vercauteren et al., 2009). More recent learning-based registration strategies use image datasets to learn a function that rapidly produces a deformation field for a given image pair (Balakrishnan et al., 2019; Rohé et al., 2017; Sokooti et al., 2017; de Vos et al., 2019; Wu et al., 2015; Yang et al., 2017). These techniques necessitate the tuning of registration hyperparameters that have dramatic impacts on the estimated deformation field. For example, optimal hyperparameter choices can differ substantially across model implementation or even image contrast and anatomy, and even small changes can have large influences on accuracy. Choosing hyperparameter values is therefore an important step in developing, testing, and distributing registration methods.

Tuning hyperparameters often involves random or grid search strategies to evaluate separate models for specific discrete hyperparameter values (Figure 1). In practice, researchers or model users typically go through an iterative process of optimizing and validating models using a small subset of hyperparameter values and repeatedly adapting this subset based on the observed results. An optimal value for each hyperparameter is usually selected based on model performance, most often determined by human evaluation or additional validation data, such as anatomical annotations. This approach necessitates considerable computational and human effort, which, in turn, may lead to suboptimal parameter choices, misleading negative results, and impeded progress, especially when researchers resort to using values from the literature that are not appropriate for their specific dataset or registration task. For example, cross-subject registration of neuroimaging data from Alzheimer's Disease patients with significant atrophy will require a substantially different optimal regularization hyperparameter than longitudinal same-subject registration, as we illustrate in our experiments.

We present HyperMorph, a markedly different strategy for tuning registration hyperparameters. Our contributions are:

Method. HyperMorph involves the end-to-end training of a single, rich model that *learns* the influence of registration hyperparameters on deformation fields, in contrast to traditional hyperparameter search (Figure 1). A HyperMorph model comprises a meta network, or a hypernetwork, that estimates a spectrum of registration models by learning a continuous function of the hyperparameters and only needs to be trained once, facilitating rapid image registration for any hyperparameter value at test-time. This avoids the need to repeatedly train a set of models for separate, fixed hyperparameters, since HyperMorph can correctly predict their outputs in substantially less computational time. Consequently, HyperMorph facilitates rapid optimization of hyperparameters for a set of validation data. This is even more important for tasks involving more than one important hyperparameter, in which the computational complexity renders traditional search strategies inadequate.

Properties. By capitalizing on the similarity of networks with similar hyperparameters, an individual HyperMorph model employs weight-sharing to optimize efficiently relative to the

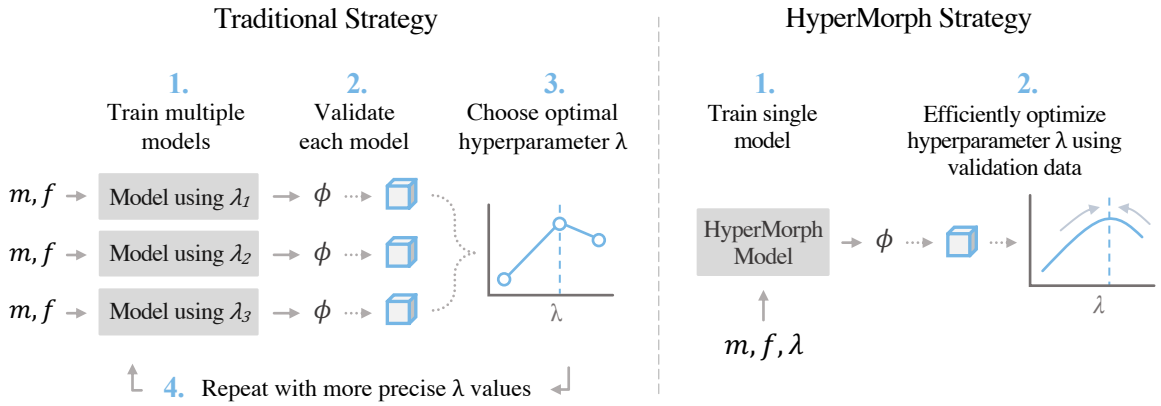

Figure 1: Traditional hyperparameter search strategies (left) involve the optimization of multiple registration models (that predict a deformation ϕ for an input image pair m, f) using different hyperparameter values (λ) and often require repeating the search for finer hyperparameter resolutions or different ranges. The HyperMorph strategy (right) trains a *single* network that approximates a landscape of traditional models and can be evaluated for any hyperparameter value at test-time.

time required to train the multiple registration models it is able to encompass. Furthermore, we demonstrate that HyperMorph registration accuracy is less variable across multiple random network initializations compared to conventional registration models, reducing the need to retrain.

Utility. HyperMorph facilitates rapid *test-time* search of optimal hyperparameter values through automatic optimization or visual evaluation for a continuous range of hyperparameters. We show the benefit of this technique by employing a *single* HyperMorph model to identify optimal hyperparameter values for different loss metrics, datasets, anatomical regions, or tasks with substantially more precision than grid search methods.

This paper extends work presented at the 2021 International Conference on Information Processing in Medical Imaging (Hoopes et al., 2021). This extension introduces and analyzes an alternative approach to learning the effect of registration hyperparameters by integrating an additional hyperparameter input within monolithic registration networks, as an alternative to using hypernetworks. We contrast the hypernetwork-based HyperMorph approach with this integrative approach. In addition, we also improve the hypernetwork-based HyperMorph architecture. In our experiments, we add additional analyses for the effect of network size and hyperparameter sampling strategy on HyperMorph accuracy, and evaluate the ability of HyperMorph to learn the effect of multiple hyperparameters in semi-supervised training using 3D images (as opposed to 2D slices). We also introduce a thorough discussion of this paradigm.

2. Related Work

In this section, we introduce the techniques and common hyperparameters involved in modern image registration, and we provide an overview of hyperparameter tuning methods and hypernetwork-based architectures in machine learning.

2.1 Image Registration

Image registration is widely studied in many formulations. Classical registration methods find a deformation field by optimizing an energy function independently for each image pair. This often involves maximizing an image-matching term that measures similarity between aligned images while enforcing a regularization on the deformation field to encourage topological correctness or smoothness on the resulting warp. Methods include B-spline based deformations (Rueckert et al., 1999), discrete optimization methods (Dalca et al., 2016; Glocker et al., 2008), elastic models (Bajcsy and Kovacic, 1989), SPM (Ashburner and Friston, 2000), LDDMM (Beg et al., 2005; Cao et al., 2005; Hernandez et al., 2009; Joshi and Miller, 2000; Miller et al., 2005; Zhang et al., 2017), symmetric normalization (Avants et al., 2008), Demons (Vercauteren et al., 2009), DARTEL (Ashburner, 2007), and spherical registration (Fischl et al., 1999). These techniques are robust and yield precise alignments, but iterative pairwise registration is typically computationally costly, often requiring tens of minutes or more to align image volumes (with size 256^3) on a CPU. More recent GPU-based implementations are faster and operate on the order of minutes or even seconds, but require access to a GPU for each registration (Brunn et al., 2021; Modat et al., 2010; Shamonin et al., 2014).

Recent learning-based approaches to registration use convolutional neural networks (CNNs) to learn a function that computes the deformation field for a given image pair in seconds on a CPU or faster on a GPU. Supervised models are trained to predict deformation fields that have been been simulated or computed by other techniques (Krebs et al., 2017; Rohé et al., 2017; Sokooti et al., 2017; Yang et al., 2017), whereas unsupervised, or self-supervised, strategies are trained end-to-end and optimize an energy function similar to classical cost functions (Balakrishnan et al., 2019; Dalca et al., 2019b; Krebs et al., 2019; Mok and Chung, 2020; de Vos et al., 2019; Zhao et al., 2019). Semi-supervised strategies leverage auxiliary information, like anatomical annotations, in the loss function to improve test registration accuracy (Balakrishnan et al., 2019; Hering et al., 2019; Hoffmann et al., 2021; Hu et al., 2018).

Commonly, these methods depend on at least one influential hyperparameter that balances the weight of the image-matching term with that of the deformation-regularization term. Semi-supervised losses might require an additional hyperparameter to weight an auxiliary term. Furthermore, the loss terms themselves often contain important hyperparameters, like the number of bins in mutual information (Viola and Wells III, 1997) or the neighborhood size (window size) of local normalized cross-correlation (Avants et al., 2011). Unfortunately, tuning hyperparameters in classical registration is an inefficient procedure since it typically requires tens of minutes to hours to compute pair-wise registrations. Although learning-based methods facilitate rapid registration at test-time, training individual models for different hyperparameter values is computationally expensive and can take days

or even weeks to converge, resulting in hyperparameter searches that consume hundreds of GPU-hours (Balakrishnan et al., 2019; Hoffmann et al., 2021; de Vos et al., 2019).

2.2 Hyperparameter Optimization

Hyperparameter tuning is a fundamental component of general learning-based model development that aims to jointly optimize a validation objective conditioned on model hyperparameters and a training objective conditioned on model weights (Franceschi et al., 2018). In common hyperparameter optimization methods, model training is considered a black-box function. Standard, popular approaches include random, grid, and sequential search (Bergstra and Bengio, 2012). More sample-efficient approaches involve Bayesian optimization techniques, which adopt a probabilistic model of the objective function to seek optimal hyperparameter values (Bergstra et al., 2011; Mockus et al., 1978; Snoek et al., 2012). These methods are often time-consuming because they require multiple model optimizations for each assessment of the hyperparameter. Various adaptations of Bayesian strategies improve efficiency by extrapolating model potential from learning curves (Domhan et al., 2015; Klein et al., 2016), prioritizing resources to promising models with bandit-based methods (Jamieson and Talwalkar, 2016; Li et al., 2017), and evaluating cheap approximations of the black-box function of interest (Kandasamy et al., 2017).

Unlike black-box methods, gradient-based hyperparameter tuning strategies compute gradients of the validation error as a function of the hyperparameters by differentiating through the nested learning procedure. Reverse-mode automatic differentiation facilitates the optimization of thousands of hyperparameters, but reversing the entire training procedure is exceedingly memory intensive (Bengio, 2000; Domke, 2012; Maclaurin et al., 2015). To conserve overhead, DrMAD (Fu et al., 2016) approximates the training procedure reversal by accounting for the parameter trajectory, and other approaches consider only the last parameter update for each optimization iteration (Luketina et al., 2016). Alternative approaches compute the hyperparameter gradient by deriving an implicit equation for the gradient under certain conditions (Pedregosa, 2016) or in real-time through forward-mode differentiation (Franceschi et al., 2017).

All of these automatic hyperparameter tuning methods require optimization for an explicit validation objective. However, a comprehensive set of annotated validation data might not be available for every registration task, and in some cases registration accuracy must be evaluated visually or through a non-differentiable downstream measure. Furthermore, hyperparameters are generally optimized once for single set of validation data, and it is not easy to modify hyperparameter values rapidly (e.g. for a new task) without retraining models.

2.3 Hypernetworks

Hypernetworks are meta neural networks that output the weights of a primary network (Ha et al., 2016; Schmidhuber, 1993), and these two networks comprise a single model that is trained end-to-end. Hypernetworks were originally introduced to compress model size (Ha et al., 2016), but they have been used in a variety of applications across other domains, including posterior estimation in Bayesian neural networks (Krueger et al., 2017; Ukai et al., 2018), automatic network pruning (Li et al., 2020; Liu et al., 2019), functional representa-

tion (Klocek et al., 2019; Spurek et al., 2020), multi-task learning (Meyerson and Miikkulainen, 2019), and generative models (Ratzlaff and Fuxin, 2019). The influence of hypernetwork initialization strategies has also been explored extensively (Chang et al., 2019).

Additionally, hypernetworks have drawn recent attention as a promising tool for gradient-based hyperparameter optimization, as they facilitate direct differentiation through the entire learning procedure with respect to the hyperparameters of interest. For example, SMASH (Brock et al., 2017) employs a hypernetwork to estimate model parameters for a given architecture. Other frameworks use hypernetworks to tune regularization hyperparameters for image classification models and demonstrate that hypernetworks are capable of approximating the overall effect of these hyperparameters (Lorraine and Duvenaud, 2018; MacKay et al., 2019). HyperMorph employs hypernetworks in the context of learning-based registration to learn how hyperparameter values impact predicted deformation fields, similar to recent work for k-space reconstruction (Wang et al., 2021). A parallel, independent work also investigates learning the effect of regularization weights in registration models. The proposed method presents a different mechanism that emphasizes conditional instance normalization (Dumoulin et al., 2016) and employs an MLP, conditioned on the regularization parameter, to shift the feature statistics of each internal feature map (Mok and Chung, 2021).

3. Methods

Registration. Deformable registration methods align a moving image m and a fixed image f by computing a correspondence ϕ. We build on unsupervised learning-based registration approaches, which establish a standard registration network $g_{\theta_g}(m, f) = \phi$, with trainable parameters θ_g, that predicts the optimal deformation ϕ for the input image pair $\{m, f\}$. The deformation map ϕ is often implemented by adding a predicted displacement field to the identity map of the n-dimensional spatial domain $\Omega \in \mathbb{R}^n$.

These models contain a variety of hyperparameters, and the underlying objective of HyperMorph is to learn the effect of *loss* hyperparameters Λ on the deformation field ϕ. We propose two fundamentally different ways of achieving this. In the first, we employ a hypernetwork to modify the registration function g_{θ_g} as a function of hyperparameters Λ. In the second, we extend the existing registration function g_{θ_g} to take in hyperparameters Λ as input. We focus our development on the former, hypernetwork-based approach, which is substantially easier to optimize and yields better results.

3.1 HyperMorph

We propose a nested registration function, in which a hypernetwork $h_{\theta_h}(\Lambda) = \theta_g$, with parameters θ_h, estimates the parameters of the primary registration network θ_g for input sample values of Λ (Figure 2). We use stochastic gradient methods to optimize hypernetwork parameters θ_h with the loss function:

$$\mathcal{L}_h(\theta_h; \mathcal{D}) = \mathbb{E}_{\Lambda \sim p(\Lambda)} \Big[\mathcal{L}(\theta_h; \mathcal{D}, \Lambda) \Big], \tag{1}$$

where $h_{\theta_h}(\Lambda) = \theta_g$, $\mathcal{D}$ is a training dataset of images, $\mathcal{L}(\cdot)$ is a registration loss function with hyperparameters Λ, and $p(\Lambda)$ is a prior probability over the hyperparameters of

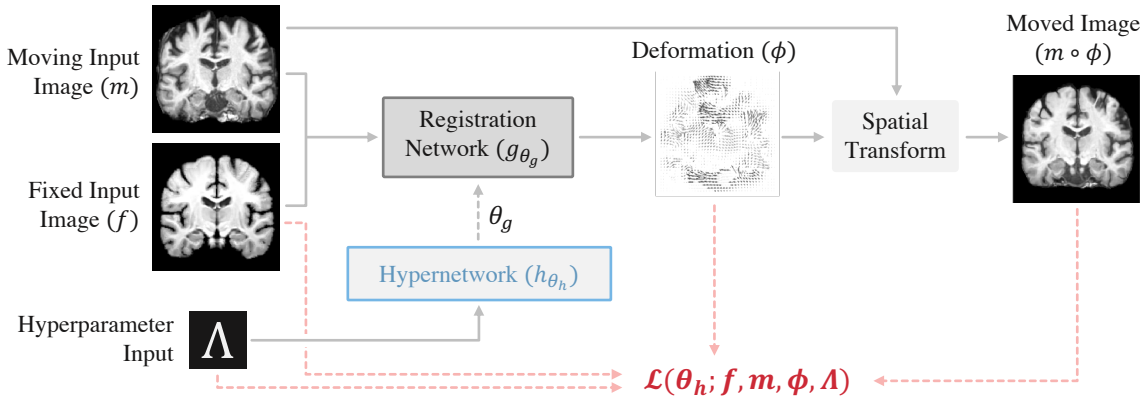

Figure 2: The HyperMorph architecture comprises a hypernetwork $h_{\theta_h}(\Lambda) = \theta_g$ that takes registration hyperparameters Λ as input and estimates the parameters of a primary registration network g_{θ_g}. HyperMorph is trained end-to-end as a single model with a continuous range of hyperparameter values, capitalizing on the implicit weight-sharing that captures the redundancy that exists amongst a landscape of registration networks.

interest. This distribution $p(\Lambda)$ can be uniform over a defined range or tailored to match assumptions. For each optimization step, we sample values from $p(\Lambda)$ and use these in the loss function $\mathcal{L}(\cdot)$ and as input to the network $h_{\theta_h}(\cdot)$. Introducing a level of abstraction, the hypernetwork h_{θ_h} allows the convolutional kernels θ_g of the registration network g_{θ_g} to flexibly adapt to varying hyperparameter values.

Unsupervised Model Instantiation. We build on unsupervised approaches to learning-based registration, which commonly involve optimizing a loss of the form:

$$\mathcal{L}(m, f, \phi; \Lambda) = \mathcal{L}_{sim}(f, m \circ \phi; \lambda_{sim}) + \lambda \mathcal{L}_{reg}(\phi; \lambda_{reg}) \tag{2}$$

where $m \circ \phi$ represents m warped by ϕ. The $\mathcal{L}_{sim}$ loss term quantifies the similarity between $m \circ \phi$ and f and includes potential hyperparameters λ_{sim}, whereas the $\mathcal{L}_{reg}$ term measures the spatial regularity of the estimated deformation field ϕ and includes potential hyperparameters λ_{reg}. The hyperparameter λ regulates the weight of $\mathcal{L}_{reg}$, and we define $\Lambda = \{\lambda, \lambda_{sim}, \lambda_{reg}\}$. One limitation of this formulation is that $p(\lambda)$ is challenging to define as the range of λ is infinite. We constrain λ to $[0, 1]$ by scaling $\mathcal{L}_{sim}$ by $(1 - \lambda)$. We thus optimize HyperMorph using:

$$\mathcal{L}_h(\theta_h; \mathcal{D}) = \mathbb{E}_\Lambda \Big[\sum_{m,f \in \mathcal{D}} \Big((1 - \lambda) \mathcal{L}_{sim}(f, m \circ \phi; \lambda_{sim}) + \lambda \mathcal{L}_{reg}(\phi; \lambda_{reg}) \Big) \Big], \tag{3}$$

where $\phi = g_{\theta_g}(m, f)$ and $\theta_g = h_{\theta_h}(\Lambda)$.

In our experiments, we use mean-squared error (MSE) and *local* normalized cross-correlation (NCC) as the similarity metrics for $\mathcal{L}_{sim}$ when registering images of the same contrast, and we use mutual information (MI) for multi-contrast registration. Local NCC

involves a hyperparameter that defines the local neighborhood (window) size, and MI involves a hyperparameter that controls the number of histogram bins (Viola and Wells III, 1997). In some cases, MSE is scaled by estimated image noise σ^{-2}.

To encourage diffeomorphic deformations, which are invertible by design, we spatially integrate the vectors of a stationary velocity field (SVF) v using *scaling and squaring* (Arsigny et al., 2006; Ashburner, 2007; Dalca et al., 2019b) to obtain ϕ, which is regularized using

$$\mathcal{L}_{reg}(\phi) = \frac{1}{2}\sum_{i=1}^{n} \|\nabla v_i\|^2, \tag{4}$$

where i is an axis in the n-dimensional image and $\nabla v_i(p)$ defines the spatial gradient of v_i at location $p \in \Omega$. The regularization term $\mathcal{L}_{reg}$ can take a variety of forms and might include multiple specific hyperparameters λ_{reg}.

Semi-supervised Model Instantiation. Following recent strategies that exploit supplemental information during training, we extend HyperMorph to semi-supervised learning by incorporating segmentation maps in the loss function:

$$\begin{aligned}\mathcal{L}_h(\theta_h; \mathcal{D}) = \mathbb{E}_\Lambda \sum_{m,f \in \mathcal{D}} \Big[&(1-\lambda)(1-\gamma)\mathcal{L}_{sim}(f, m \circ \phi; \lambda_{sim}) \\ &+ \lambda \mathcal{L}_{reg}(\phi; \lambda_{reg}) + \gamma \mathcal{L}_{seg}(s_f, s_m \circ \phi)\Big],\end{aligned} \tag{5}$$

where s_m and s_f are segmentation maps corresponding to the moving and fixed images, respectively, and $\mathcal{L}_{seg}$ is a measure of segmentation overlap, often the Dice coefficient (Dice, 1945), weighted by the hyperparameter γ. As with the unsupervised loss, we constrain the range of γ within $[0, 1]$ by scaling the similarity term $\mathcal{L}_{sim}$ by $(1-\lambda)(1-\gamma)$.

3.2 Hyperparameter Tuning

An optimized HyperMorph model can rapidly register a test image pair $\{m, f\}$ as a function of important hyperparameters. If external annotation data is not available, hyperparameters may be efficiently tuned in an interactive fashion. In some cases, landmarks, functional data, or segmentation maps are present, facilitating fast automatic hyperparameter optimization for a validation dataset.

Interactive. Users can manually adjust hyperparameter values in close to real-time using interactive sliders until they are visually satisfied with the alignment of a given image pair. Sometimes, the user might adopt different settings when focusing on particular domains of the image. For instance, the optimal value of the λ hyperparameter, which balances image-similarity and regularization, can differ substantially across anatomical regions of the brain (see Figure 10). Interactive tuning is feasible since HyperMorph can efficiently estimate the influence of λ values on the deformation field ϕ without necessitating further training.

Automatic. If additional information, such as segmentation maps $\{s_m, s_f\}$, are present for validation, an individual trained HyperMorph model facilitates rapid optimization of

hyperparameter values using

$$\Lambda^* = \arg\min_{\Lambda} \mathcal{L}(\Lambda; \theta_h, \mathcal{D}, \mathcal{V}) = \arg\min_{\Lambda} \sum_{\substack{m,f \in \mathcal{D} \\ s_m, s_f \in \mathcal{V}}} \mathcal{L}_{val}(s_f, s_m \circ \phi), \tag{6}$$

where $\mathcal{V}$ is a set of validation segmentations and $\mathcal{L}_{val}$ is a validation loss to be minimized. To carry out this hyperparameter optimization, we freeze the hypernetwork parameters θ_h so that the input Λ represents the sole set of trainable parameters. We rapidly optimize (6) using stochastic gradient descent strategies.

3.3 Implementation

We implement HyperMorph with the open-source VoxelMorph registration library (Balakrishnan et al., 2019), modeling the base registration network g_{θ_g} with a U-Net-like architecture (Ronneberger et al., 2015). In our experiments, this comprises a four-layer convolutional encoder-like part, with 16, 32, 32, and 32 respective channels per layer, followed by a seven-layer convolutional decoder-like part, with 32, 32, 32, 32, 32, 16, and 16 respective channels per layer. The convolutional layers have a kernel size of 3, a stride of 1, and are activated using LeakyReLU with α parameter 0.2. After each convolution in the encoder, we reduce the spatial dimensions using max pooling with a window size of 2, and in the decoder, each convolution is followed by an upsampling layer until the volume is returned to full resolution. Skip connections concatenate features of the encoder with features of the first decoder layer of equal resolution. A final, linearly-activated convolutional layer outputs an SVF, which is integrated with five scaling and squaring steps to obtain ϕ (Ashburner, 2007; Dalca et al., 2019b). In total, this base model g_{θ_g} contains 313,507 trainable parameters.

In the hypernetwork-based HyperMorph models used throughout our experiments, h_{θ_h} consists of five fully-connected layers, each with 32, 64, 64, 128, and 128 respective units and ReLu activations, followed by a final linearly-activated layer with output units corresponding to the number of trainable parameters in g_{θ_g}. This is improved from the previous HyperMorph implementation (Hoopes et al., 2021), which yielded slightly worse accuracy compared to some baseline models and used a hypernetwork consisting of four fully-connected layers, each with 64 units and a tanh-activated final layer. Together, the registration network and hypernetwork constitute a single network with approximately 40.5 million trainable parameters (θ_h) that exist entirely in the hypernetwork. Since the large majority of trainable parameters exist in the final layer of the hypernetwork, the model size increases substantially with the number of parameters in g_{θ_g}, but this increase does not lead to substantial memory footprint, as this is dominated by the convolutional tensors. We emphasize that the proposed strategy pertains to any learning-based registration architecture, not just VoxelMorph.

We train all HyperMorph and baseline models with the Adam optimizer (Kingma and Ba, 2014), a batch size of 1, and an initial learning rate of 10^{-4}, employing a decay strategy that reduces the learning rate by a factor of two for every 5×10^4 optimization steps without improvement in the training loss. Continuous hyperparameter values are randomly sampled from a uniform distribution during training. Based on our experiments, the agreement of HyperMorph with baselines at the boundary hyperparameter values $\{0, 1\}$ of λ

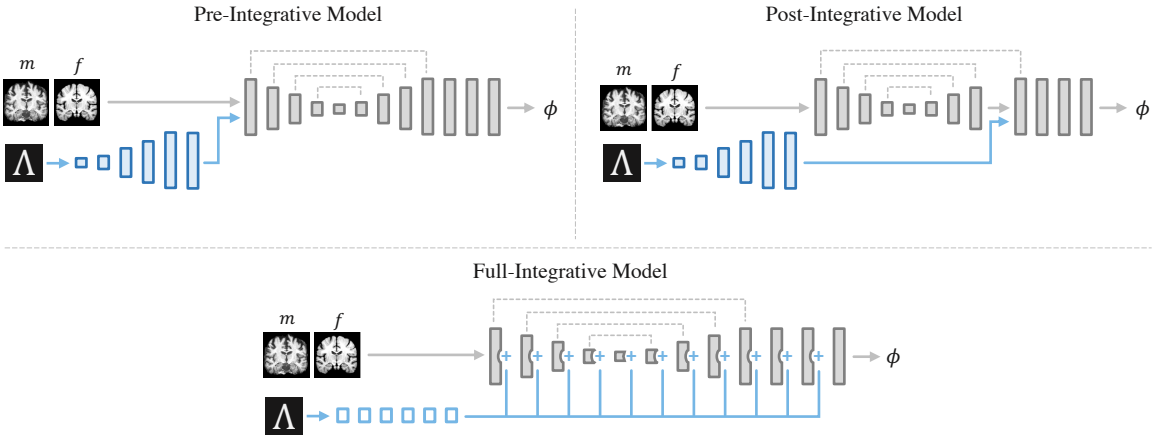

Figure 3: Alternative model architectures for learning the effect of registration hyperparameters. In these approaches, the hyperparameters Λ are provided as input to an auxiliary convolutional network d_{θ_d} (blue), which is integrated directly with the primary registration network g_{θ_g} (grey). The output of d_{θ_d} is either added to the output channels of the registration U-Net (full-integrative model) or provided as an additional input to the first layer (pre-integrative model) or last upsampling layer (post-integrative model).

can be improved if values are slightly over-sampled during training. We let r be the fraction of hyperparameters sampled from this end-point distribution $\lambda \in \{0, 1\}$ and set it to 0.2 in our experiments. Discrete hyperparameters, like the local NCC window size, are sampled from a *discrete* uniform distribution during training, and we normalize the sampled values in this range to $[0, 1]$ when used as input to the hypernetwork. We observe that HyperMorph learns a continuous function for these hyperparameters by interpolating weights across discrete values, enabling their direct optimization at test-time using gradient strategies. We implement HyperMorph in Python, using the TensorFlow (Abadi et al., 2016) and Keras (Chollet et al., 2015) packages, and release HyperMorph as a component of the broader VoxelMorph registration package, with plans to support a PyTorch (Paszke et al., 2019) implementation. We train and evaluate all models on Nvidia Quadro RTX 8000 GPUs.

3.4 Alternative Models

We also analyze a fundamentally different approach to amortized hyperparameter learning by extending the *inputs* to the registration function as opposed to changing the registration function using hypernetworks. We build on architectures that combine scalar or non-image inputs with convolutional networks used in other tasks, such as probabilistic segmentation (Kohl et al., 2018) or conditional template construction (Dalca et al., 2019a).

We examine three alternative implementations, in which hyperparameters are provided as input to a small auxiliary convolutional sub-network d_{θ_d}, with parameters θ_d, that is joined directly with the primary registration network (Figure 3). In the first two alternative

Table 1: Three groups of image datasets are used throughout the experiments and split into train, validate, and test subsets of specified size.

Group	Train	Validate	Test	Datasets
Within-contrast	7,400	5,000	5,030	ABIDE (Di Martino et al., 2014) ADHD-200 (Milham et al., 2012) GSP (Dagley et al., 2017) MCIC (Gollub et al., 2013) OASIS-1 (Marcus et al., 2007) PPMI (Marek et al., 2011) UK Biobank (Sudlow et al., 2015) Buckner40 (Fischl et al., 2002)
Multi-contrast	496	496	496	HCP (Bookheimer et al., 2019) FSM (in-house data)
Longitudinal	48	48	48	OASIS-2 (Marcus et al., 2010)

architectures, input hyperparameters are repeated and reshaped to an $8\times8\times8$ multi-channel volume, with one channel for each input hyperparameter, and provided as input to a series of six convolutional layers in d_{θ_d}, each with 32 channels. The output of each layer in d_{θ_d} is upsampled until the target image resolution is reached. In the first alternative architecture, the pre-integrative network, g_{θ_g} takes the output of d_{θ_d} as an additional input (Dalca et al., 2019a). In the second architecture, the post-integrative network, the output of d_{θ_d} is concatenated with the input to the *final* upsampling layer of the U-Net (Kohl et al., 2018). In the third alternative architecture, d_{θ_d} comprises five fully-connected layers, each with 256 units and ReLu activations, followed by a linearly-activated layer with output units equal to the total number of channels across all layers in g_{θ_g}. We refer to this architecture as the full-integrative network, and each value estimated by d_{θ_d} is added to its corresponding convolutional output channel in the base network.

4. Experiments

We conduct experiments evaluating how well a single HyperMorph model captures the behavior and matches the performance of individually trained registration networks with separate hyperparameter values. We show that our approach substantially reduces the computational and human effort required for a search with one or two registration hyperparameters. We present considerable improvements in robustness to model initialization. We also illustrate the utility of HyperMorph for efficient hyperparameter optimization across different subpopulations of data, image contrasts, registration types, and individual anatomical structures. Additionally, we compare hypernetwork-based HyperMorph with the proposed alternative models that expand the input space, and we provide a framework analysis exploring the effect of hypernetwork size and hyperparameter sampling on HyperMorph performance.

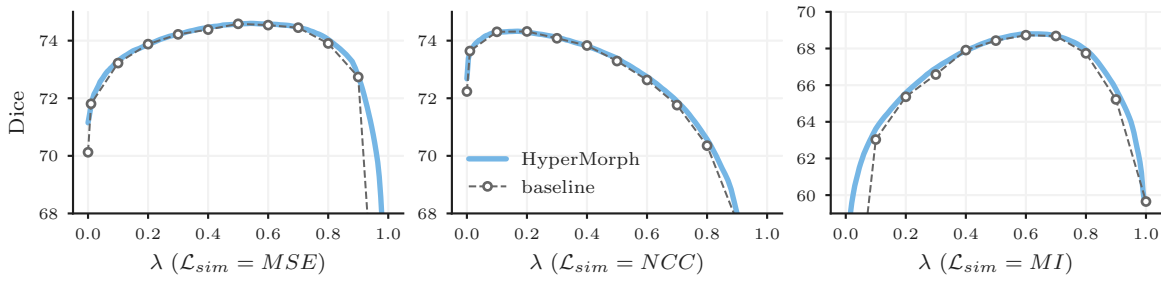

Figure 4: Mean Dice scores achieved by a single HyperMorph model (blue) and baselines trained for different regularization weights λ (grey) when using MSE, NCC, or MI similarity metrics.

Datasets. We use three groups of 3D brain magnetic resonance imaging (MRI) data gathered across multiple public datasets, as summarized in Table 1. The first group includes a series of within-contrast T1-weighted (T1w) scans, and the second group is a multi-contrast collection of T1w and T2-weighted (T2w) images, FLASH scans acquired with various flip angles, and MPRAGE scans with different inversion times. We also employ a group of longitudinal images for comparisons between within-subject and cross-subject registration tasks, in which we consider two T1w scans, acquired at least one year apart for each individual.

Using FreeSurfer 7.2 (Fischl, 2012), all MR images are resampled as 256×256×256 volumes with 1-mm isotropic resolution, bias-corrected, brain-extracted, and automatically anatomically segmented for evaluation. We affinely align all images to the FreeSurfer Talairach atlas and uniformly crop them to size 160×192×224. When evaluating registration accuracy with segmentation data, we consider standard anatomical labels provided by FreeSurfer: the thalamus, caudate, putamen, pallidum, hippocampus, amygdala, accumbens area, ventral diencephalon, choroid plexus, cerebral cortex, cerebral white matter, cerebellar cortex, cerebellar white matter, brain stem, cerebrospinal fluid, and the 3rd, 4th, and lateral ventricles. FreeSurfer generates accurate segmentations that are often considered a silver-standard for automatic brain labeling (Dalca et al., 2019c; Puonti et al., 2016), but we also employ an auxiliary set of 30 manually-labeled T1w images from the Buckner40 cohort to evaluate registration accuracy using gold-standard annotations. This dataset was not used during training.

Evaluation metrics. For evaluation, we compute the volumetric Dice overlap coefficient (reported as percentage points between 0 and 100) as well as the 95th percentile surface distance in millimeters for corresponding anatomical labels of the moved and fixed segmentation maps. To quantify regularity of the deformation ϕ, we report the standard deviation of the Jacobian determinant $|J_\phi|$, where $J_\phi(p) = \nabla\phi(p)$ for each displacement voxel $p \in \Omega$.

Baseline Models. HyperMorph can be applied to any learning-based registration architecture. To analyze how accurately it captures the effect of hyperparameters on the inner registration network $g_{\theta_g}(\cdot)$, we train baseline VoxelMorph models with architectures identical to $g_{\theta_g}(\cdot)$, each with a different set of fixed hyperparameter values.

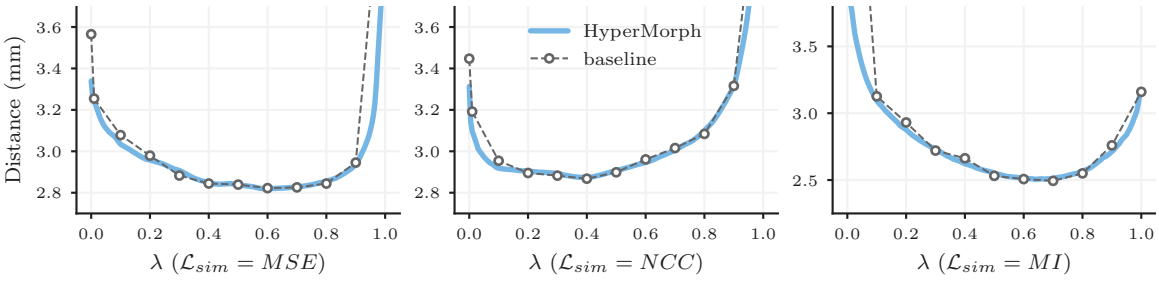

Figure 5: Mean 95th percentile surface distances achieved by a single HyperMorph model (blue) and baselines trained for different regularization weights λ (grey) when using MSE, NCC, or MI similarity metrics.

4.1 Experiment 1: HyperMorph Efficiency and Capacity

The goal of this experiment is to assess the extent to which a single HyperMorph model captures a landscape of baseline models trained with different hyperparameter values. We emphasize that we do not focus on comparing HyperMorph with the latest registration architecture but rather on evaluating how HyperMorph can be combined with any framework.

Setup. We first compare the accuracy and computational cost of a single HyperMorph model to standard grid hyperparameter search for the regularization weight λ. In separate analyses, we train HyperMorph and the VoxelMorph baselines using the MSE ($\sigma = 0.05$) and NCC (window size $= 9^3$) similarity metrics for within-contrast registration, as well as the MI metric (32 fixed bins) for cross-contrast registration. For each metric, we train 12 baseline models and compare network performance across 50 randomly selected image pairs from the test set. To analyze HyperMorph in the context of domain-shift scenarios, we further evaluate models (trained with $\mathcal{L}_{sim} = MSE$) on 20 image pairs from the manually-labeled Buckner40 cohort, held out entirely from training.

Additionally, we assess the ability of HyperMorph to learn the effect of multiple hyperparameters simultaneously. First, we train a semi-supervised HyperMorph model using a subset of six labels, holding out a further six labels for evaluation, to simulate partially annotated data. In this experiment, we choose λ and the relative weight γ of the semi-supervised loss (5) as the hyperparameters of interest. Second, we train a HyperMorph model treating λ and the local NCC window size w as hyperparameters. Since the computation of local NCC is computationally prohibitive for large window sizes in 3D data, we conduct the experiment in 2D on mid-coronal slices. These slices are not bias-corrected during preprocessing, since the local NCC metric is most useful for aligning images with strong intensity inhomogeneities. We train semi-supervised baseline models for 25 hyperparameter combinations, performing a discrete search on a 5×5 two-dimensional grid.

Results. *Computational Cost.* A single HyperMorph model converges considerably faster than the baseline grid search. For single-hyperparameter tests, HyperMorph requires 6.1 times fewer GPU-hours than the 1D grid search with 12 baseline models (Table 2). For two hyperparameters, the difference is even more striking, with HyperMorph requiring 12.3 times fewer GPU-hours than a grid search with 25 baseline models. Furthermore, a 5×5 grid

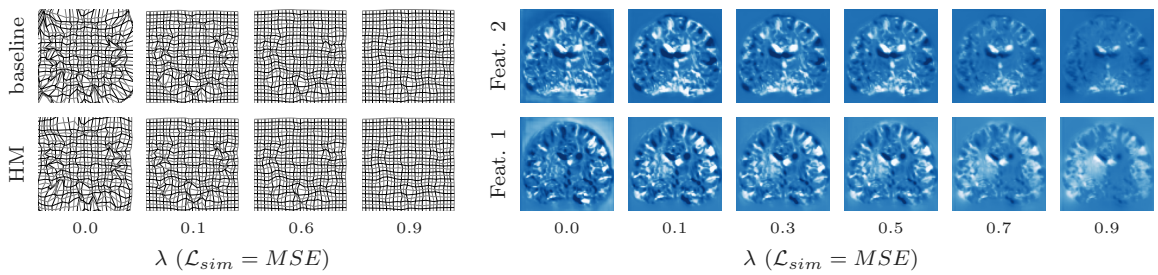

Figure 6: Left: visual comparison of HyperMorph (HM) and baseline model registration deformations on a mesh grid, illustrating similar changes in regularity across λ values. Right: representative changes in feature activations of the final layer in the HyperMorph U-Net for different regularization weights.

search is coarse, especially if the scale of the evaluated hyperparameters is unknown. While the time required for grid search is proportional to the number of grid points, HyperMorph enables arbitrarily fine resolution between grid points, at no increase in training time.

Representation accuracy. Along with the computational advantage, Figures 4, 5, 7, and 8 show that HyperMorph yields optimal hyperparameter values similar to those determined through the baseline-model grid search. For each image pair, an average difference in the optimal hyperparameter value λ^* of only 0.04 ± 0.06 across single-hyperparameter experiments results in a negligible maximum Dice difference of 0.06 ± 0.42 (on a scale of 0 to 100) and a minimum surface distance of 0.01 ± 0.02 millimeters. Even when evaluated on the held-out, manually-labeled dataset, HyperMorph similarly matches the baseline registration accuracy, differing in maximum Dice by 0.02 ± 0.09 and in minimum surface distance by 0.01 ± 0.03 millimeters (Figure S1). Furthermore, the deformation field regularity at λ^*, measured by standard deviation of the Jacobian determinant, is 0.31 ± 0.14 and differs by only 0.01 ± 0.01 across HyperMorph and baseline models. We visualize these deformation fields and HyperMorph channel activations in Figures 6 and S2.

Semi-supervised experiments yield a maximum Dice difference of only 0.02 ± 0.27 and minimum surface distance of 0.01 ± 0.01. Figure 8 showcases an example in which the optimal pair of $\{\lambda, w\}$ values identified by HyperMorph lies far from the points of the coarse search

Table 2: Total train time (left) and model variability across random initializations (right) for HyperMorph and baseline grid search techniques. Train time for the 2 hyp. (λ, w) experiment is substantially faster as it was conducted using 2D image slices as opposed to 3D volumes.

	Train time (total GPU-hours)			**Variability (SD)**	
	1 hyp. (λ)	2 hyp. (λ, γ)	2 hyp. (λ, w)	MSE	MI
HyperMorph	**192.5 ± 23.1**	**321.9 ± 16.1**	**4.0 ± 0.1**	**0.100**	**0.127**
Baseline	1,174.9 ± 196.1	4,120.5 ± 295.4	46.8 ± 5.7	0.176	0.325

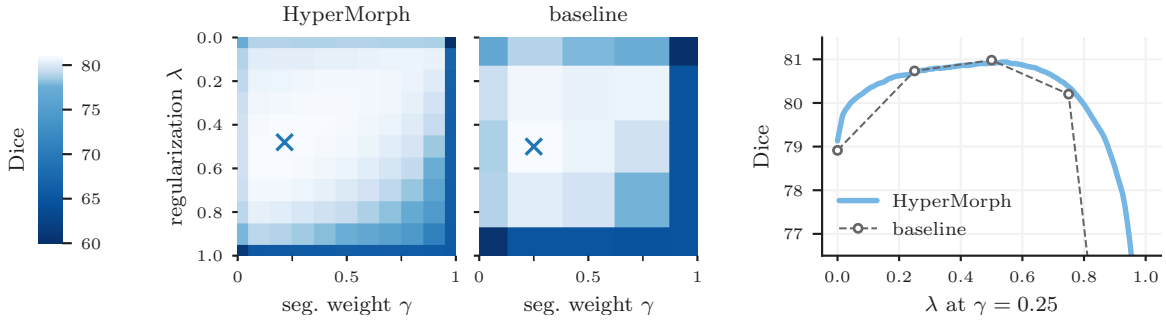

Figure 7: Two-dimensional hyperparameter search for semi-supervised registration with regularization hyperparameter λ and segmentation weight γ. For a set of 50 test pairs, the cross markers indicate the optimal λ, γ values determined by HyperMorph and a baseline grid search. We compute total Dice using both sets of training and held-out labels. While the left hyperparameter space is illustrated on a discrete grid for visualization, HyperMorph enables evaluating the effect of hyperparameter values at arbitrarily fine resolution.

grid, resulting in a 0.78 ± 0.98 decrease in maximum Dice for the traditional approach. In practice, even fewer baselines might be trained for a coarser hyperparameter search, resulting in either suboptimal hyperparameter choice or sequential search with substantial manual overhead.

4.2 Experiment 2: Robustness to Initialization

The goal of this experiment is to analyze the robustness of each hyperparameter search strategy to different network weight initialization.

Setup. We repeat the previous single-hyperparameter experiment with MSE and MI, retraining five HyperMorph models from scratch. For each of four different λ values, we also train five baseline models. Each training run re-initializes the kernel weights with a different randomization seed, and we compare the variability across initializations in terms of the standard deviation (SD) of Dice accuracy for the HyperMorph and baseline networks, in a set of 50 image pairs.

Results. Figure 9 shows that HyperMorph is considerably more robust (lower SD) to differential initialization than the baselines. Across the entire range of λ, the average Dice SD for HyperMorph models trained with MSE is 1.7 times lower ($P < .001$ via paired t-test) than the baseline SD and 2.6 times lower for MI ($P < .001$) (Table 2).

4.3 Experiment 3: Hyperparameter-Tuning Utility

This experiment aims to validate HyperMorph as a powerful tool for hyperparameter tuning across a number of registration tasks, with or without annotated validation data.

Setup. *Interactive Tuning.* We demonstrate the utility of HyperMorph through an interactive tool that enables visual optimization of hyperparameters even if no annotated

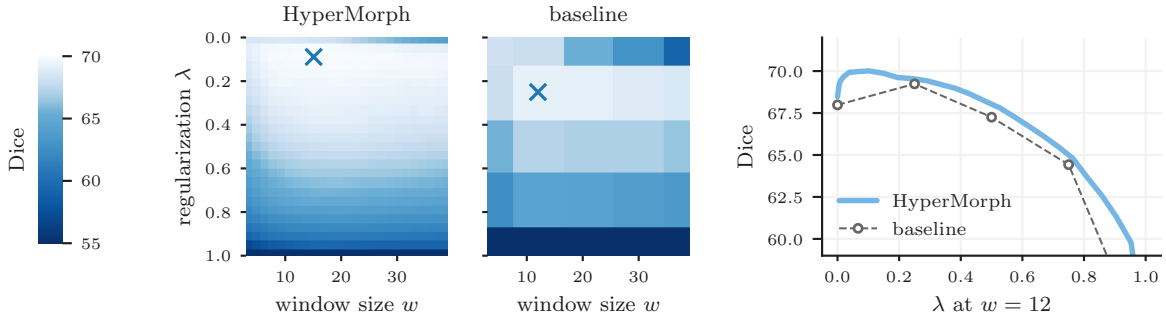

Figure 8: Two-dimensional hyperparameter search for unsupervised registration with regularization weight λ and local NCC window size w. For a set of 50 test pairs, the cross markers indicate the optimal λ, w values determined by HyperMorph and a baseline grid search. HyperMorph is able to identify the optimal λ, w value pair missed by a traditional grid search.

data are available. The user can explore the effect of *continuously varying* hyperparameter values using a single trained model and manually select a preferred optimal deformation. We provide an interactive HyperMorph demonstration with associated code at `http://hypermorph.voxelmorph.net`.

Automatic Tuning. When annotations are available for validation, we can efficiently optimize the hyperparameter λ in an automated fashion. For a variety of applications, we identify the optimal regularization weight λ^* for sets of 50 registration pairs. First, we investigate how λ^* differs across subject subpopulations and anatomical regions: we train HyperMorph on a subset of our T1w training data, and optimize λ separately for sets of ABIDE, GSP, MCIC, and UK Biobank (UKB) subjects at test time. With this single HyperMorph model, we also identify separate values of λ^* for a range of neuroanatomical regions. Second, we train HyperMorph on a subset of the multi-contrast image pairs and determine λ^* separately for T1w-to-T2w, T2w-to-T2w, and multi-flip-angle (multi-FA) registration tasks. Last, we analyze the extent to which λ^* differs between cross-sectional and longitudinal registration: we train HyperMorph on a combination of within-subject and cross-subject pairs from OASIS-2 and separately optimize λ for test pairs within and across subjects.

Results. Figure 10 shows that λ^* varies substantially across subpopulations, image contrasts, tasks, and anatomical regions. Importantly, in some cases using the λ^* computed for one subset of data on another results in considerably reduced accuracy. For example, using λ^* determined for GSP on ABIDE data decreases the maximum attainable Dice score by 1.86 ± 2.87. We hypothesize that the observed variability in optimal hyperparameter values is caused by differences in image quality and anatomy between the datasets. Similarly, using the multi-FA λ^* for T1w-to-T2w registration and the within-subject λ^* for cross-subject registration causes the respective maximum Dice scores to drop by 3.16 ± 2.14 and 1.73 ± 1.20. Lastly, Figure 10D illustrates that the optimal λ value varies broadly across anatomical regions, suggesting that it is desirable to choose regularization weights depending on the downstream task and focus of a given study. In our experiments, automatic

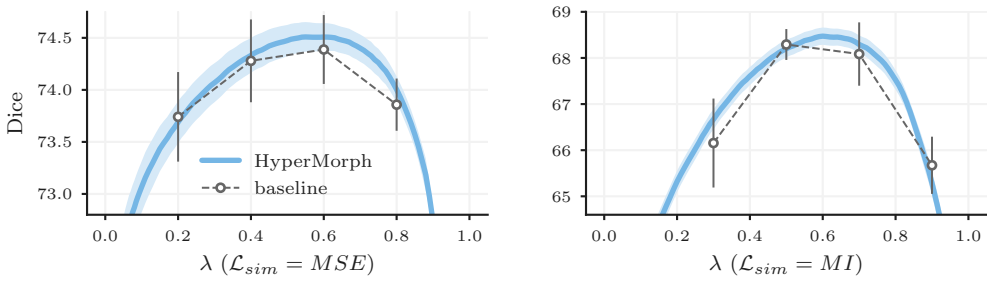

Figure 9: Variability across several training initializations for HyperMorph and baseline models. Error bars and fill regions span a 2-σ range around the mean registration accuracy, which is substantially tighter for HyperMorph.

hyperparameter optimization takes just 12.3 ± 1.8 seconds on average per test pair and requires 10 GB of memory, while interactive tuning requires only 2 GB. We emphasize that these metrics are influenced substantially by the size of the image data being registered.

4.4 Experiment 4: Hypernetwork Size

We also measure the importance of hypernetwork capacity for accurate representation of individually trained baseline models.

Setup. We train separate HyperMorph models for three hypernetwork sizes: small (with 16, 16, 16, and 16 units per layer), medium (with 32, 32, 64, 64, and 64 units per layer), and large (with 32, 64, 64, 128, and 128 units per layer). We carry out these and all subsequent experiments using MSE for $\mathcal{L}_{sim}$ and evaluate model accuracy against baselines results for 50 image pairs.

Results. Figure 11A shows that the capability of HyperMorph to match baseline registration accuracy increases with hypernetwork size. The large hypernetwork is appropriate for learning the effect of the regularization weight λ in 3D registration. Although the large hypernetwork contains approximately 7.6 times more trainable weights than the small network, we find no substantial difference ($< 0.4\%$) in total training or inference time across hypernetwork sizes, likely because the significant bottleneck is caused by convolutional operations.

4.5 Experiment 5: Hyperparameter Sampling

This experiment evaluates how different hyperparameter sampling methods affect HyperMorph accuracy. In previous tests, we observe that sampling regularization weights λ from a uniform distribution during HyperMorph training results in registration accuracy comparable to baseline models across most of the hyperparameter range, especially near λ^*, but less comparable estimations very close to the boundaries $\lambda \in \{0, 1\}$, corresponding to similarity-only or regularization-only loss functions.

Setup. To investigate whether these boundaries can also be captured by HyperMorph, we over-sample the end-point values $\{0, 1\}$ of the hyperparameter λ at a fixed rate r. We

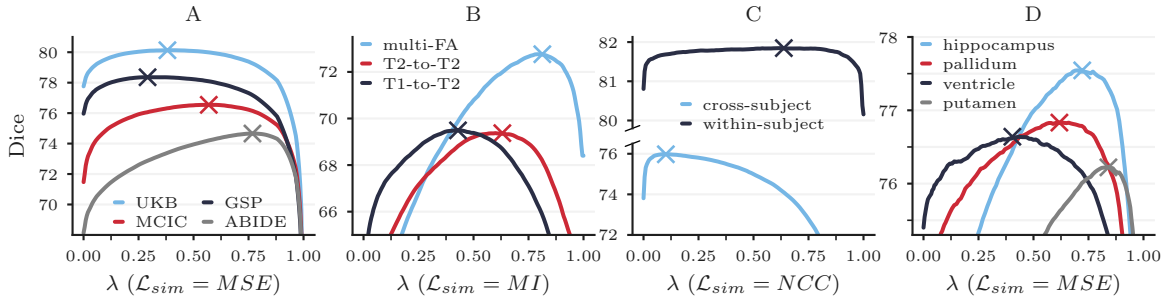

Figure 10: Registration accuracy across dataset subpopulations (A), image contrasts (B), tasks (C), and neuroanatomical regions (D). The cross markers indicate the optimal value λ^* as identified by automatic hyperparameter optimization.

train and evaluate three separate models for different values of r (0.0, 0.2, and 0.8) and compare the final accuracy against baselines, to asses the influence of this rate on registration accuracy.

Results. HyperMorph models trained for large values of r closely match the expected registration accuracy at end-point values of λ but sacrifice registration accuracy across all values of λ (Figure 11B). For example, when training HyperMorph with $r = 0.0$ (no over-sampling), the mean deviation from the baseline Dice is 0.08 ± 0.26 at λ^*, compared to 2.96 ± 1.57 at $\lambda \in \{0, 1\}$. However, with $r = 0.8$, the mean deviation from baseline Dice is 0.87 ± 0.40 at λ^* and 0.49 ± 0.51 at $\lambda \in \{0, 1\}$. We emphasize that over-sampling is only necessary to estimate appropriate representations at the extreme hyperparameter boundaries. As similarity-only or regularization-only loss functions are not desirable for the majority of applications, uniform sampling will suffice in most cases. Throughout all experiments presented in this study, we choose an intermediate value of $r = 0.2$, which facilitates the most consistent matching of baseline registration accuracy for all values of λ.

4.6 Experiment 6: Alternative Models

While hypernetworks facilitate learning the effect of hyperparameters on registration networks, we also investigate the alternative HyperMorph strategy of adding an input to the standard registration network.

Setup. We train the pre-integrative, post-integrative, and full-integrative architectures defined in Section 3.4 and compare the resulting registration accuracy and computational cost with the individual baseline models.

Results. None of the three alternative networks yield the accuracy achieved by baseline and hypernetwork-based HyperMorph models (Figure 11C). While the pre-integrative and full-integrative networks broadly encapsulates the effect of λ, identifying the baseline λ^* with an average error of 0.07 ± 0.08 and 0.05 ± 0.07, respectively, they deviate from peak baseline accuracy by 0.48 ± 0.25 and 0.23 ± 0.46. The post-integrative network struggles to learn the accurate effect of λ, deviating from the baseline λ^* by 0.19 ± 0.12 and peak accuracy by 0.71 ± 0.61. The total train time for the full-integrative model is $1.1\times$ longer than that of

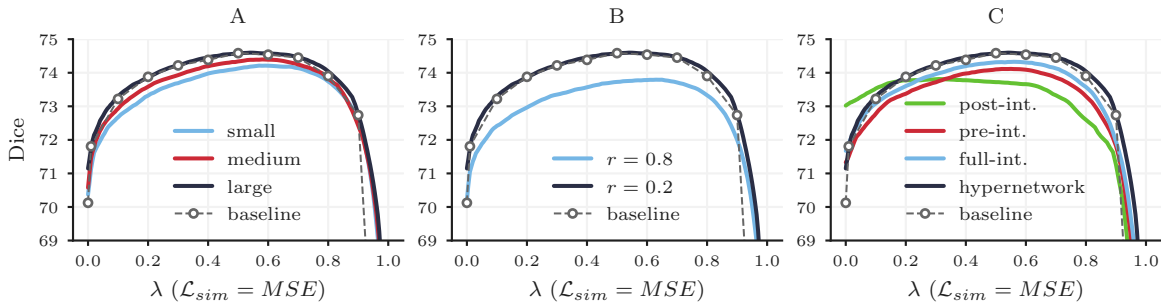

Figure 11: Analysis showing registration accuracy in terms of Dice overlap for HyperMorph models trained with different hypernetwork sizes (A), end-point sampling rates r (B), and HyperMorph strategies (C).

the hypernetwork-based HyperMorph, while the pre-integrative and post-integrative models require 1.8× more time, likely due to the added convolutional operations in the network.

5. Discussion and Conclusion

The accuracy of learning-based deformable registration algorithms largely hinges on the choice of adequate hyperparameter values, which might differ substantially across registration targets, data types, model architectures, and loss implementations. Consequently, accurate and high-resolution hyperparameter search is an essential component of registration model development.

In this work, we present HyperMorph, a learning strategy for registration that eliminates the need to repeatedly train the same model with different hyperparameter values to evaluate their effect on performance. HyperMorph employs a hypernetwork that takes the desired hyperparameter values as input and predicts the corresponding parameters, or weights, of a registration network. We show that training a *single* HyperMorph model is sufficient to capture the behavior of a range of baseline models individually optimized for different hyperparameter values. This enables *precise* hyperparameter optimization at test-time, because the optimal value may be located between the limited number of discrete grid points evaluated by traditional approaches.

We explore two alternatives for choosing optimum regularization weights: one interactive, based jointly on image matching and visual smoothness, and one automatic, based on registration accuracy. The automatic method optimizes Dice overlap, which in itself does not take field regularity into account. We ensure that this parameterization yields regular deformations by analyzing voxel-wise Jacobian determinants. However, we emphasize that HyperMorph enables efficient optimization of hyperparameter values at test-time using any desired metric of choice.

Function vs. Input Space. We explore two paradigms for learning the effect of registration hyperparameters on the deformation field: a hypernetwork-based function that returns an appropriate registration function given a hyperparameter value or a modification of the registration function to accept an addition hyperparameter value as input (pre-integrative,

post-integrative, or full-integrative). In the analysis, the latter approach under-performs in registration quality, and thus, modelling the effect of hyperparameters in this manner presents a more challenging optimization problem. We hypothesize that this effect could be due to the fact that the convolutional filters are fixed once training is complete, requiring them to perform a substantially more difficult task than simple registration. In contrast, the hypernetwork approach enables the convolutional filters to flexibly adapt to specific hyperparameter values, suggesting a more powerful mechanism. Further analysis of these effects is an interesting future direction but is beyond the scope of this work.

We emphasize that a hypernetwork is not the only effective mechanism for learning the effects of hyperparameters on registration networks, and we investigate this group of alternative architectures in an attempt to gain and provide insight across approaches. For example, parallel, independent work (Mok and Chung, 2021) explores conditional registration networks. These learn regularization effects by leveraging instance normalization and employing an MLP to scale and shift hidden features as a function of the regularization weight λ. This strategy is similar to the full-integrative implementation, suggesting another promising alternative strategy. The approach is also similar to hypernetwork-based HyperMorph since it employs an MLP to learn the hyperparameter effect, but it differs in how this MLP is coupled with the registration network. It is likely that with sufficient architectural optimization, both hypernetworks and specifically-designed conditional CNNs are powerful solutions for a variety of hyperparameter learning tasks.

Computational efficiency. By exploiting the similarity of networks across a range of hyperparameter values, a single HyperMorph model uses weight-sharing to efficiently learn to estimate optimal deformation fields for arbitrary image pairs and *any* hyperparameter value from a continuous interval. This enables fast, automated tuning of hyperparameters at test time and represents a substantial advantage over traditional search techniques: to identify an optimal configuration, these techniques typically optimize a number of registration networks across a sparse, discrete grid of hyperparameter values, which requires dramatically more compute and human time than HyperMorph.

Initialization robustness. Experiment 2 demonstrates that HyperMorph is substantially more robust to network weight initialization than individually trained networks, exhibiting 43 to 61% reduced variability over training runs, likely because the combined hypernetwork and registration-network stack can take advantage of weight-sharing across a landscape of hyperparameter values. This result further underlines the computational efficiency provided by HyperMorph, since traditional tuning approaches often resort to training models multiple times at each grid point to remove potential bias due to initialization variability.

Test-time adaptation. Existing registration models are often trained using a single hyperparameter value optimized globally for a set of validation data. However, the frequently overlooked reality is that hyperparameter optima can differ substantially across individual image pairs and applications, whereas most, if not all, registration-based analysis pipelines assume the existence of a single optimal hyperparameter value (Patenaude et al., 2011; Wang et al., 2012; Fischl, 2012). For example, a pair of images with very different anatomies would benefit from weak regularization, permitting warps of high non-linearity. This implies that learning-based methods capable of adapting hyperparameters on the fly are essential. We demonstrate that a single HyperMorph model enables rapid discovery of optimal hyperpa-

rameter values for different dataset subpopulations, image contrasts, registration tasks, and even individual anatomical regions, facilitating the future development of models that learn to estimate ideal hyperparameter values for individual registration pairs.

Further work. HyperMorph can be used with hyperparameters beyond those evaluated in this work. For example, it could be applied to the number of bins in the MI metric, the choice of form of the regularization term λ_{reg}, the hyperparameters used in the regularization term(s), the level of dropout, or even architectural hyperparameters, similarly to the SMASH method (Brock et al., 2017). However, the effects of certain hyperparameters, especially those related to model architecture, might be substantially more difficult for a hypernetwork to learn. Additionally, for HyperMorph to learn the effects of some optimization-specific parameters, like learning rate and batch size, it would likely require substantial modifications.

The identification of λ^* for different brain regions promotes a potential future direction of estimating a spatially-varying field of regularization hyperparameters for simultaneously optimal registration of all anatomical structures. Additionally, while we evaluate HyperMorph for one and two hyperparameters, we expect this strategy to readily adapt to more hyperparameters and are eager to explore hypernetworks in this context, in which grid search is impractical. We are also interested in investigating how the benefits of implicit weight-sharing in hypernetworks might differ across categories of loss hyperparameters.

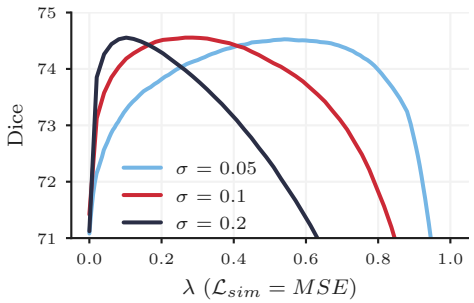

Figure 12: Registration accuracy (Dice) achieved by HyperMorph models trained for different values of estimated image noise σ^{-2}.

We also plan to expand this work by exploring more complex distributions of $p(\Lambda)$ and how they affect hyperparameter search. For example, in registration formulations where the image similarity term is re-weighted by estimated image noise σ^{-2}, the range of the hyperparameter space that should be searched can vary substantially. With a suboptimal choice of σ, a grid search is often even more challenging, as the range of hyperparameter values that perform well can be very narrow. In a preliminary experiment, we found that HyperMorph performed well for a variety of noise estimates σ, even with a uniform distribution $p(\lambda) = \mathcal{U}(0, 1)$ used throughout our experiments (Figure 12). However, the result also simultaneously highlights the more dramatic Dice score sensitivity to hyperparameter choice for some σ values, suggesting that non-uniform distributions might lead to even better HyperMorph performance.

Conclusion. We believe HyperMorph has the potential to drastically alleviate the burden of retraining networks with different hyperparameter values, thereby enabling efficient development of finely optimized models for image registration. While the training strategy described in this paper is well-suited for tuning a visually-driven workflow like image registration, the technique can be used to improve other applications within and beyond the domain of medical imaging analysis.

Acknowledgments

Support for this research was provided in part by the BRAIN Initiative Cell Census Network (U01 MH117023), the National Institute for Biomedical Imaging and Bioengineering (P41 EB015896, 1R01 EB023281, R01 EB006758, R21 EB018907, R01 EB019956, P41 EB030006), the National Institute on Aging (1R56 AG064027, 1R01 AG064027, 5R01 AG008122, R01 AG016495, 1R01 AG070988), the National Institute of Mental Health (R01 MH123195, R01 MH121885, 1RF1 MH123195), the National Institute for Neurological Disorders and Stroke (R01 NS0525851, R21 NS072652, R01 NS070963, R01 NS083534, 5U01 NS086625, 5U24 NS10059103, R01 NS105820), the NIH Blueprint for Neuroscience Research (5U01 MH093765), the multi-institutional Human Connectome Project, the National Institute of Child Health and Human Development (K99 HD101553), and the Wistron Corporation. This research was made possible through resources provided by Shared Instrumentation Grants 1S10 RR023401, 1S10 RR019307, and 1S10 RR023043.

Ethical Standards

The work follows the highest ethical standards in conducting research and writing the manuscript. All models were trained using publicly available data, with the exception of the FSM data. We believe that HyperMorph can substantially improve the use of image registration in medical image analysis, and we use the UK Biobank data in accordance with this interest in public health.

Conflicts of Interest

Bruce Fischl has a financial interest in CorticoMetrics, and his interests are reviewed and managed by Massachusetts General Hospital and Mass General Brigham in accordance with their conflict of interest policies.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

Vincent Arsigny, Olivier Commowick, Xavier Pennec, and Nicholas Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 924–31. Springer, 2006.

J. Ashburner. A fast diffeomorphic image registration algorithm. *Neuroimage*, 38(1):95–113, 2007.

John Ashburner and Karl J Friston. Voxel-based morphometry-the methods. *Neuroimage*, 11:805–821, 2000.

Brian B Avants, Charles L Epstein, Murray Grossman, and James C Gee. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical Image Analysis*, 12(1):26–41, 2008.

Brian B Avants, Nicholas J Tustison, Gang Song, Philip A Cook, Arno Klein, and James C Gee. A reproducible evaluation of ants similarity metric performance in brain image registration. *Neuroimage*, 54(3):2033–2044, 2011.

R. Bajcsy and S. Kovacic. Multiresolution elastic matching. *Computer Vision, Graphics, and Image Processing*, 46:1–21, 1989.

Guha Balakrishnan, Amy Zhao, Mert R Sabuncu, John Guttag, and Adrian V Dalca. Voxelmorph: a learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38(8):1788–1800, 2019.

M Faisal Beg, Michael I Miller, Alain Trouvé, and Laurent Younes. Computing large deformation metric mappings via geodesic flows of diffeomorphisms. *International Journal of Computer Vision*, 61(2):139–157, 2005.

Yoshua Bengio. Gradient-based optimization of hyperparameters. *Neural Computation*, 12 (8):1889–1900, 2000.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(1):281–305, 2012.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2011.

Susan Y Bookheimer, David H Salat, Melissa Terpstra, Beau M Ances, Deanna M Barch, Randy L Buckner, Gregory C Burgess, Sandra W Curtiss, Mirella Diaz-Santos, Jennifer Stine Elam, et al. The lifespan human connectome project in aging: an overview. *NeuroImage*, 185:335–348, 2019.

Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Smash: one-shot model architecture search through hypernetworks. *arXiv preprint arXiv:1708.05344*, 2017.

Malte Brunn, Naveen Himthani, George Biros, Miriam Mehl, and Andreas Mang. Fast gpu 3d diffeomorphic image registration. *Journal of Parallel and Distributed Computing*, 149: 149–162, 2021.

Yan Cao, Michael I Miller, Raimond L Winslow, and Laurent Younes. Large deformation diffeomorphic metric mapping of vector fields. *IEEE Transactions on Medical Imaging*, 24(9):1216–1230, 2005.

Oscar Chang, Lampros Flokas, and Hod Lipson. Principled weight initialization for hypernetworks. In *International Conference on Learning Representations*, 2019.

François Chollet et al. Keras. `https://keras.io`, 2015.

Alexander Dagley, Molly LaPoint, Willem Huijbers, Trey Hedden, Donald G McLaren, Jasmeer P Chatwal, Kathryn V Papp, Rebecca E Amariglio, Deborah Blacker, Dorene M Rentz, et al. Harvard aging brain study: dataset and accessibility. *NeuroImage*, 144: 255–258, 2017.

Adrian Dalca, Marianne Rakic, John Guttag, and Mert Sabuncu. Learning conditional deformable templates with convolutional networks. *Advances in neural information processing systems*, 32, 2019a.

Adrian V Dalca, Andreea Bobu, Natalia S Rost, and Polina Golland. Patch-based discrete registration of clinical brain images. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 60–67. Springer, 2016.

Adrian V. Dalca, Guha Balakrishnan, John Guttag, and Mert Sabuncu. Unsupervised learning of probabilistic diffeomorphic registration for images and surfaces. *Medical Image Analysis*, 57:226–236, 2019b.

Adrian V Dalca, Evan Yu, Polina Golland, Bruce Fischl, Mert R Sabuncu, and Juan Eugenio Iglesias. Unsupervised deep learning for bayesian brain mri segmentation. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 356–365. Springer, 2019c.

Bob D de Vos, Floris F Berendsen, Max A Viergever, Hessam Sokooti, Marius Staring, and Ivana Išgum. A deep learning framework for unsupervised affine and deformable image registration. *Medical Image Analysis*, 52:128–143, 2019.

Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6):659–667, 2014.

Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26 (3):297–302, 1945.

Tobias Domhan, Jost Tobias Springenberg, and Frank Hutter. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

Justin Domke. Generic methods for optimization-based modeling. In *AISTATS*, pages 318–326, 2012.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

Bruce Fischl, Martin I Sereno, Roger BH Tootell, and Anders M Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human brain mapping*, 8(4):272–284, 1999.

Bruce Fischl, David H Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Van Der Kouwe, Ron Killiany, David Kennedy, Shuna Klaveness, et al. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002.

Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pages 1165–1173. PMLR, 2017.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimilano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. *arXiv preprint arXiv:1806.04910*, 2018.

Jie Fu, Hongyin Luo, Jiashi Feng, Kian Hsiang Low, and Tat-Seng Chua. Drmad: distilling reverse-mode automatic differentiation for optimizing hyperparameters of deep neural networks. *arXiv preprint arXiv:1601.00917*, 2016.

Ben Glocker, Nikos Komodakis, Georgios Tziritas, Nassir Navab, and Nikos Paragios. Dense image registration through mrfs and efficient linear programming. *Medical Image Analysis*, 12(6):731–741, 2008.

Randy L Gollub, Jody M Shoemaker, Margaret D King, Tonya White, Stefan Ehrlich, Scott R Sponheim, Vincent P Clark, Jessica A Turner, Bryon A Mueller, Vince Magnotta, et al. The mcic collection: a shared repository of multi-modal, multi-site brain image data from a clinical investigation of schizophrenia. *Neuroinformatics*, 11(3):367–388, 2013.

David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.

Alessa Hering, Sven Kuckertz, Stefan Heldmann, and Mattias P Heinrich. Enhancing label-driven deep deformable image registration with local distance metrics for state-of-the-art cardiac motion tracking. In *Bildverarbeitung für die Medizin 2019*, pages 309–314. Springer, 2019.

Monica Hernandez, Matias N Bossa, and Salvador Olmos. Registration of anatomical images using paths of diffeomorphisms parameterized with stationary vector field flows. *IJCV*, 85(3):291–306, 2009.

Malte Hoffmann, Benjamin Billot, Juan E Iglesias, Bruce Fischl, and Adrian V Dalca. Learning mri contrast-agnostic registration. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 899–903. IEEE, 2021.

Andrew Hoopes, Malte Hoffmann, Bruce Fischl, John Guttag, and Adrian V Dalca. Hypermorph: Amortized hyperparameter learning for image registration. In *International Conference on Information Processing in Medical Imaging*, pages 3–17. Springer, 2021.

Yipeng Hu, Marc Modat, Eli Gibson, Wenqi Li, Nooshin Ghavami, Ester Bonmati, Guotai Wang, Steven Bandula, Caroline M Moore, Mark Emberton, et al. Weakly-supervised convolutional neural networks for multimodal image registration. *Medical Image Analysis*, 49:1–13, 2018.

Kevin Jamieson and Ameet Talwalkar. Non-stochastic best arm identification and hyperparameter optimization. In *AISTATS*, pages 240–248, 2016.

Sarang C Joshi and Michael I Miller. Landmark matching via large deformation diffeomorphisms. *IEEE Transactions on Image Processing*, 9(8):1357–1370, 2000.

Kirthevasan Kandasamy, Gautam Dasarathy, Jeff Schneider, and Barnabás Póczos. Multi-fidelity bayesian optimisation with continuous approximations. *arXiv preprint arXiv:1703.06240*, 2017.

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

Aaron Klein, Stefan Falkner, Jost Tobias Springenberg, and Frank Hutter. Learning curve prediction with bayesian neural networks. In *ICLR: International Conference on Learning Representations*, 2016.

Sylwester Klocek, Łukasz Maziarka, Maciej Wołczyk, Jacek Tabor, Jakub Nowak, and Marek Śmieja. Hypernetwork functional image representation. In *International Conference on Artificial Neural Networks*, pages 496–510. Springer, 2019.

Simon Kohl, Bernardino Romera-Paredes, Clemens Meyer, Jeffrey De Fauw, Joseph R Ledsam, Klaus Maier-Hein, SM Eslami, Danilo Jimenez Rezende, and Olaf Ronneberger. A probabilistic u-net for segmentation of ambiguous images. *Advances in neural information processing systems*, 31, 2018.

Julian Krebs, Tommaso Mansi, Hervé Delingette, Li Zhang, Florin C Ghesu, Shun Miao, Andreas K Maier, Nicholas Ayache, Rui Liao, and Ali Kamen. Robust non-rigid registration through agent-based action learning. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 344–352. Springer, 2017.

Julian Krebs, Hervé Delingette, Boris Mailhé, Nicholas Ayache, and Tommaso Mansi. Learning a probabilistic model for diffeomorphic registration. *IEEE Transactions on Medical Imaging*, 38(9):2165–2176, 2019.

David Krueger, Chin-Wei Huang, Riashat Islam, Ryan Turner, Alexandre Lacoste, and Aaron Courville. Bayesian hypernetworks. *arXiv preprint arXiv:1710.04759*, 2017.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(1):6765–6816, 2017.

Yawei Li, Shuhang Gu, Kai Zhang, Luc Van Gool, and Radu Timofte. Dhp: Differentiable meta pruning via hypernetworks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16*, pages 608–624. Springer, 2020.

Zechun Liu, Haoyuan Mu, Xiangyu Zhang, Zichao Guo, Xin Yang, Kwang-Ting Cheng, and Jian Sun. Metapruning: Meta learning for automatic neural network channel pruning.

In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3296–3305, 2019.

Jonathan Lorraine and David Duvenaud. Stochastic hyperparameter optimization through hypernetworks. *arXiv preprint arXiv:1802.09419*, 2018.

Jelena Luketina, Mathias Berglund, Klaus Greff, and Tapani Raiko. Scalable gradient-based tuning of continuous regularization hyperparameters. In *ICML*, pages 2952–2960, 2016.

Matthew MacKay, Paul Vicol, Jon Lorraine, David Duvenaud, and Roger Grosse. Self-tuning networks: Bilevel optimization of hyperparameters using structured best-response functions. *arXiv preprint arXiv:1903.03088*, 2019.

Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *ICML*, pages 2113–2122, 2015.

Daniel S Marcus, Tracy H Wang, Jamie Parker, John G Csernansky, John C Morris, and Randy L Buckner. Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults. *Journal of cognitive neuroscience*, 19(9):1498–1507, 2007.

Daniel S Marcus, Anthony F Fotenos, John G Csernansky, John C Morris, and Randy L Buckner. Open access series of imaging studies: longitudinal mri data in nondemented and demented older adults. *Journal of cognitive neuroscience*, 22(12):2677–2684, 2010.

Kenneth Marek, Danna Jennings, Shirley Lasch, Andrew Siderowf, Caroline Tanner, Tanya Simuni, Chris Coffey, Karl Kieburtz, Emily Flagg, Sohini Chowdhury, et al. The parkinson progression marker initiative (ppmi). *Progress in neurobiology*, 95(4):629–635, 2011.

Elliot Meyerson and Risto Miikkulainen. Modular universal reparameterization: Deep multi-task learning across diverse domains. *Advances in Neural Information Processing Systems*, 32, 2019.

Michael P Milham, Damien Fair, Maarten Mennes, Stewart HMD Mostofsky, et al. The adhd-200 consortium: a model to advance the translational potential of neuroimaging in clinical neuroscience. *Frontiers in systems neuroscience*, 6:62, 2012.

Michael I Miller, M Faisal Beg, Can Ceritoglu, and Craig Stark. Increasing the power of functional maps of the medial temporal lobe by using large deformation diffeomorphic metric mapping. *PNAS*, 102(27):9685–9690, 2005.

Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of bayesian methods for seeking the extremum. *Towards Global Optimization*, 2(117-129):2, 1978.

Marc Modat, Gerard R Ridgway, Zeike A Taylor, Manja Lehmann, Josephine Barnes, David J Hawkes, Nick C Fox, and Sébastien Ourselin. Fast free-form deformation using graphics processing units. *Computer methods and programs in biomedicine*, 98(3):278–284, 2010.

Tony CW Mok and Albert Chung. Conditional deformable image registration with convolutional neural network. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 35–45. Springer, 2021.

Tony CW Mok and Albert CS Chung. Large deformation diffeomorphic image registration with laplacian pyramid networks. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 211–221, 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32, 2019.

Brian Patenaude, Stephen M Smith, David N Kennedy, and Mark Jenkinson. A bayesian model of shape and appearance for subcortical brain segmentation. *Neuroimage*, 56(3): 907–922, 2011.

Fabian Pedregosa. Hyperparameter optimization with approximate gradient. *arXiv preprint arXiv:1602.02355*, 2016.

Oula Puonti, Juan Eugenio Iglesias, and Koen Van Leemput. Fast and sequence-adaptive whole-brain segmentation using parametric bayesian modeling. *NeuroImage*, 143:235–249, 2016.

Neale Ratzlaff and Li Fuxin. Hypergan: A generative model for diverse, performant neural networks. In *International Conference on Machine Learning*, pages 5361–5369. PMLR, 2019.

Marc-Michel Rohé, Manasi Datar, Tobias Heimann, Maxime Sermesant, and Xavier Pennec. Svf-net: Learning deformable image registration using shape matching. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 266–274. Springer, 2017.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 234–241. Springer, 2015.

Daniel Rueckert, Luke I Sonoda, Carmel Hayes, Derek LG Hill, Martin O Leach, and David J Hawkes. Nonrigid registration using free-form deformation: Application to breast mr images. *IEEE Transactions on Medical Imaging*, 18(8):712–721, 1999.

Jürgen Schmidhuber. A self-referential weight matrix. In *International Conference on Artificial Neural Networks*, pages 446–450, 1993.

Denis Shamonin, Esther Bron, Boudewijn Lelieveldt, Marion Smits, Stefan Klein, and Marius Staring. Fast parallel image registration on cpu and gpu for diagnostic classification of alzheimer's disease. *Frontiers in Neuroinformatics*, 7:50, 2014.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

Hessam Sokooti, Bob de Vos, Floris Berendsen, Boudewijn PF Lelieveldt, Ivana Išgum, and Marius Staring. Nonrigid image registration using multi-scale 3d convolutional neural networks. In *MICCAI: Medical Image Computing and Computer Assisted Interventions*, pages 232–239. Springer, 2017.

Przemysław Spurek, Sebastian Winczowski, Jacek Tabor, Maciej Zamorski, Maciej Zieba, and Tomasz Trzciński. Hypernetwork approach to generating point clouds. *arXiv preprint arXiv:2003.00802*, 2020.

Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, et al. Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLOS Medicine*, 12(3):e1001779, 2015.

Kenya Ukai, Takashi Matsubara, and Kuniaki Uehara. Hypernetwork-based implicit posterior estimation and model averaging of cnn. In *Asian Conference on Machine Learning*, pages 176–191. PMLR, 2018.

Tom Vercauteren, Xavier Pennec, Aymeric Perchant, and Nicholas Ayache. Diffeomorphic demons: Efficient non-parametric image registration. *NeuroImage*, 45(1):S61–S72, 2009.

Paul Viola and William M Wells III. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 24(2):137–54, 1997.

Alan Q Wang, Adrian V Dalca, and Mert R Sabuncu. Hyperrecon: Regularization-agnostic cs-mri reconstruction with hypernetworks. *Machine Learning for Medical Image Reconstruction*, pages 3–13, 2021.

Hongzhi Wang, Jung W Suh, Sandhitsu R Das, John B Pluta, Caryne Craige, and Paul A Yushkevich. Multi-atlas segmentation with joint label fusion. *IEEE transactions on pattern analysis and machine intelligence*, 35(3):611–623, 2012.

Guorong Wu, Minjeong Kim, Qian Wang, Brent C Munsell, and Dinggang Shen. Scalable high-performance image registration framework by unsupervised deep feature representations learning. *IEEE Transactions on Biomedical Engineering*, 63(7):1505–1516, 2015.

Xiao Yang, Roland Kwitt, Martin Styner, and Marc Niethammer. Quicksilver: Fast predictive image registration – a deep learning approach. *NeuroImage*, 158:378–396, 2017.

Miaomiao Zhang, Ruizhi Liao, Adrian V Dalca, Esra A Turk, Jie Luo, P Ellen Grant, and Polina Golland. Frequency diffeomorphisms for efficient image registration. In *IPMI: Information Processing in Medical Imaging*, pages 559–570. Springer, 2017.

Shengyu Zhao, Yue Dong, Eric I Chang, Yan Xu, et al. Recursive cascaded networks for unsupervised medical image registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10600–10610, 2019.

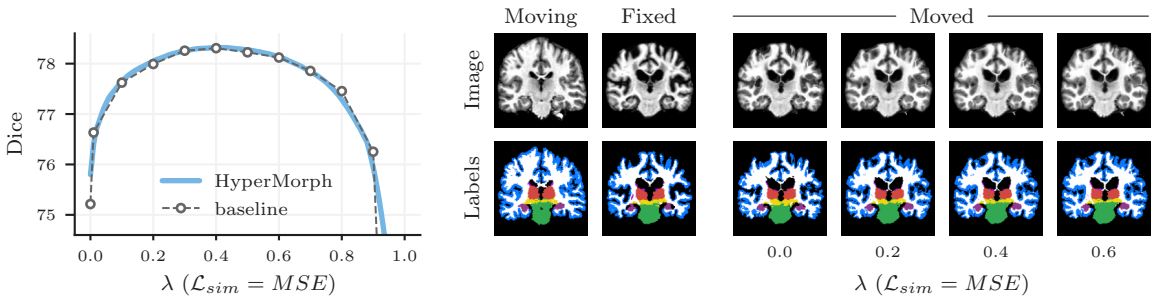

Figure S1: Left: mean Dice scores achieved by a single HyperMorph model and baselines evaluated on the held-out, manually-labeled Buckner40 dataset. Right: image and label-based qualitative changes in HyperMorph alignment across different regularization weights for a given subject pair.

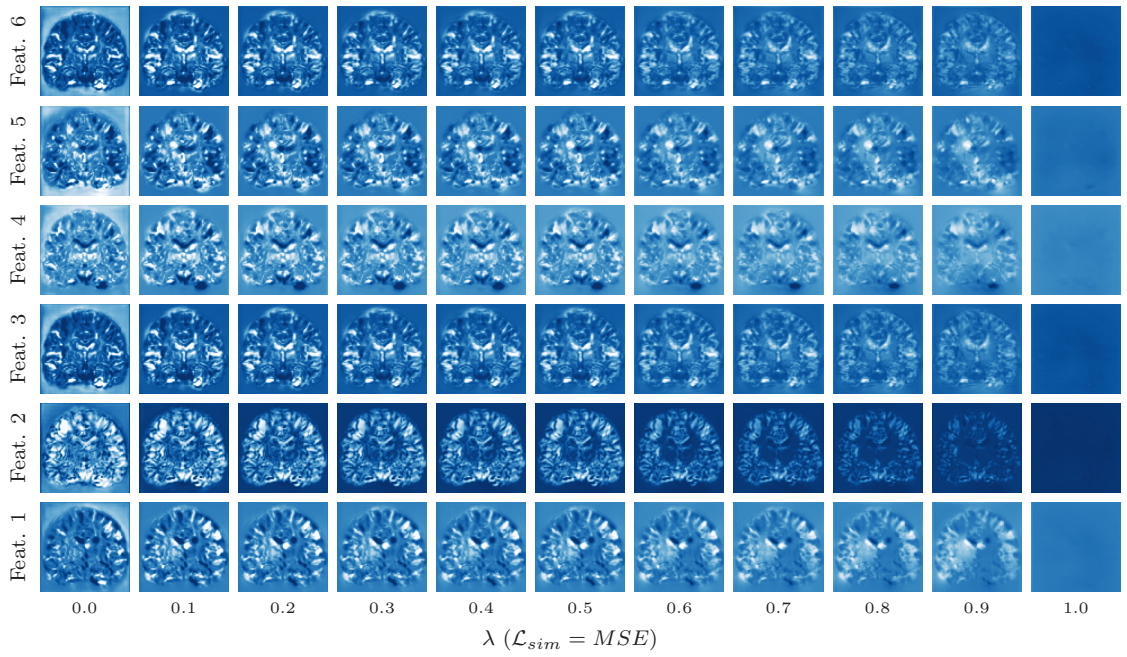

Figure S2: Changes in feature activations of the final HyperMorph U-Net layer across different values for λ.