Deformation-Invariant Sparse Coding

by

George H. Chen

B.S. with dual majors in Electrical Engineering and Computer Sciences,
Engineering Mathematics and Statistics, UC Berkeley, May 2010

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science
in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

June 2012

Signature of Author: ______________________________

George H. Chen
Department of Electrical Engineering and Computer Science
May 23, 2012

Certified by: ______________________________

Polina Golland
Associate Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: ______________________________

Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Committee for Graduate Students

Deformation-Invariant Sparse Coding

by George H. Chen

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of
Master of Science

Abstract

Sparse coding represents input signals each as a sparse linear combination of a set of basis or dictionary elements where sparsity encourages representing each input signal with a few of the most indicative dictionary elements. In this thesis, we extend sparse coding to allow dictionary elements to undergo deformations, resulting in a general probabilistic model and accompanying inference algorithm for estimating sparse linear combination weights, dictionary elements, and deformations.

We apply our proposed method on functional magnetic resonance imaging (fMRI) data, where the locations of functional regions in the brain evoked by a specific cognitive task may vary across individuals relative to anatomy. For a language fMRI study, our method identifies activation regions that agree with known literature on language processing. Furthermore, the deformations learned by our inference algorithm produce more robust group-level effects than anatomical alignment alone.

Thesis Supervisor: Polina Golland
Title: Associate Professor of Electrical Engineering and Computer Science

Acknowledgements

The past two years have been a tumultuous success, thanks to an incredible cast of people who've helped me along the way. At the forefront of this cast is Polina Golland, who has been a phenomenal advisor. Polina has provided me insight into a slew of problems, and I've lost count of how many times I've crashed into a road block, she'd suggest a way to visualize data for debugging, and—voilà—road block demolished! At the same time, Polina has been tremendously chill, letting me toil at my own pace while I sporadically derail myself with side interests divergent from the Golland group. Thanks Polina, and thanks for ensuring that I don't derail myself too much!

This thesis would not have seen the light of day without the assistance of my collaborators Ev Fedorenko and Nancy Kanwisher. As I have no background in neuroscience, Ev and Nancy were instrumental in explaining some of the basics and offered invaluable comments, feedback, and data on the neuroscience application that drives this thesis. It really was the neuroscience application that led to the main model proposed in this thesis rather than the other way around.

Meanwhile, landing in Polina's group was a culture shock in itself as my background was not in medical imaging. Luckily, members of the group have been more than accommodating. Without the patience of former group members Danial Lashkari, Gabriel Tobon, and Michal Depa in answering my barrage of questions on functional magnetic resonance imaging and image registration during my first year, my transition into working in medical imaging would have taken substantially longer with at least a tenfold increase in headaches, futility, and despair. I would also like to thank the rest of the group members for far too many delightful conversations and for putting up with my recurring bouts of frustration and fist shaking. Numerous thanks also go out to visiting group members René Donner, who hosted me in Austria, and Prof. Georg Langs, whose bicycle that I'm still borrowing has immensely improved my quality of life.

Many students outside of Polina's group have also had a remarkable impact on my education and crusade against grad-student-depressionitis. Eric Trieu has been a repository of great conversations, puns, and medical advice. James Saunderson has been my encyclopedia for math and optimization. As for machine learning, I have yet to meet any grad student who has amassed more machine learning knowledge and big-picture understanding than Roger Grosse, who persistently is rewarding to bounce ideas off of. I also want to thank the rest of the computer vision and Stochastic Systems Group students for all the productive discussions as well as new ways to procrastinate.

To solidify my understanding of probabilistic graphical models, I had the fortune of being a teaching assistant for Devavrat Shah and Greg Wornell, who were an absolute riot to work with. Too often, their perspectives on problems led me to all sorts of new intuitions and epiphanies. Many thanks also go to my co-TA's Roger Grosse (again) and George Tucker, who contributed to my project of making Khan Academy style videos for the students. I still can't believe we threw together all those videos and churned out the first ever complete set of typeset lecture notes for 6.438!

Outside of academics, I've had the pleasure and extraordinary experience of being an officer in the Sidney Pacific Graduate Residence. I would like to thank all the other officers who helped me serve the full palate of artsy, marginally educational, and downright outrageous competitive dorm events throughout the 2011-2012 academic year, and all the residents who participated, not realizing that they were actually helpless test subjects. My only hope is that I didn't accidentally deplete all the dorm funds. I would especially like to thank Brian Spatocco for being an exceptional mentor and collaborator, helping me become a far better event organizer than I imagined myself being at the start of grad school.

I end by thanking my brother and my parents for their support over the years.

Contents

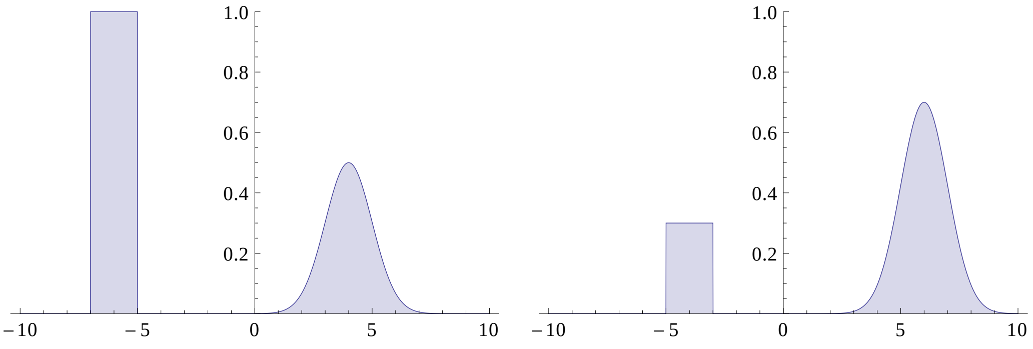
List of Figures

Chapter 1

Introduction

FINDING succinct representations for signals such as images and audio enable us to glean high-level features in data. For example, an image may be represented as a sum of a small number of edges or patches, and the presence of certain edges and patches may be used as features for object recognition. As another example, given a household's electricity usage over time, representing this signal as a sum of contributions from different electrical devices could allow us to pinpoint the culprits for a high electricity bill. These scenarios exemplify *sparse coding*, which refers to representing an input signal as a sparse linear combination of basis or *dictionary* elements, where sparsity selects the most indicative dictionary elements that explain our data. The focus of this thesis is on estimating these dictionary elements and, in particular, extending sparse coding to allow dictionary elements to undergo potentially nonlinear deformations.

To illustrate what we seek to achieve with our proposed model, we provide the following toy example. Suppose we observe the two signals shown below:



(a) Signal 1

(b) Signal 2

Figure 1.1: Toy example observed signals.

We imagine these were generated by including a box and a Gaussian bump except that they have different heights and the signal has been shifted left or right. If we don't actually know that the true shapes are a box and a Gaussian bump and we want to estimate these shapes, then a naive approach is to make an estimate based on the average of the observed signals:

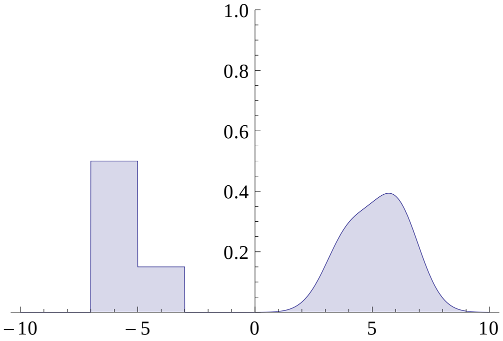
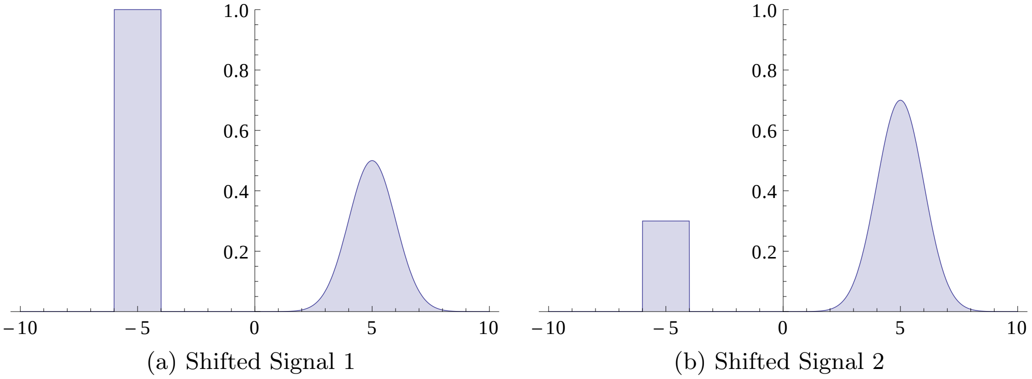


Figure 1.2: Toy example average signal.

For example, we could estimate the two underlying shapes to be the two-box mixture and the two-Gaussian-bump mixture shown above, which unfortunately don't resemble a single box and a single Gaussian bump. We could instead first align the observed signals to obtain the following shifted signals:



(a) Shifted Signal 1

(b) Shifted Signal 2

Figure 1.3: Toy example observed signals that have undergone shifts.

Then the average of these shifted signals looks as follows:

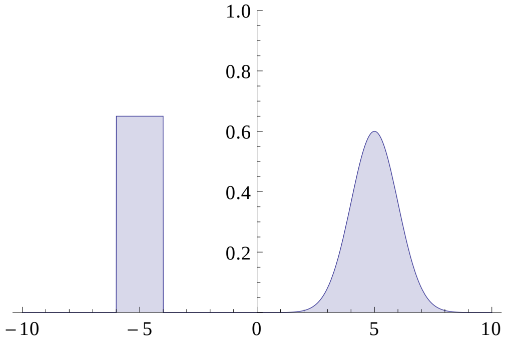


Figure 1.4: Toy example average of the shifted signals.

From the average of the shifted signals, we can recover the box and the Gaussian bump! Moreover, the peak values in the box and the Gaussian bump are, respectively, higher than the peak values in the two-box mixture and the two-Gaussian-bump mixture in Fig. 1.2, which can be viewed as a result of destructive interference in the case where we don't align the signals before averaging. Generalizing from this toy example, this thesis looks at the problem of taking as input a set of images and producing as output a dictionary (e.g., a box and a Gaussian bump in the above example) and an ensemble of deformations (which could be much more complicated then shifts) to better align the images.

The key motivating application driving this thesis is the fundamental problem in neuroscience of understanding functional organization of the brain. Mapping out where different functions, such as language processing and face recognition, evoke activations in the brain provides insight into how we as a species perform day-to-day tasks and how abnormalities in these functional locations relate to neurological disorders. But arriving at any such population-level theory of functional organization of the brain demands that we find correspondences between activation patterns evoked by a specific function across different people's brains. We cast this problem of finding correspondences between functional activation regions across individuals as a sparse coding problem where we want dictionary elements to correspond to group-level *functional units* in the brain, which refer to brain regions consistently activated by a specific task across individuals.

The problem with just applying sparse coding without incorporating deformations and hoping that the dictionary elements correspond to group-level functional units is twofold. First, people's brains vary anatomically, so images of different people's brains don't line up perfectly. However, even if we account for this anatomical variability by first pre-aligning the brains to be in the same common anatomically-normalized space, when given the same stimulus such as a sentence to read, different people's brains will exhibit activations in different locations in the normalized space! This problem of spatial variability of functional activation patterns suggests that a possible solution is to model functional units as dictionary elements that deform into the space of each individual's brain. This leads us naturally to *deformation-invariant* sparse coding, where we estimate dictionary elements that may undergo deformations, so each dictionary element is unique up to a deformation. Of course, these deformations can't be too drastic, deforming, say, a disk into any arbitrary shape.

The main contributions of this thesis are as follows:

- We formulate a probabilistic model for deformation-invariant sparse coding and provide an accompanying inference algorithm that alternates between estimating sparse linear combination weights, deformations, and dictionary elements. For estimating each deformation, the inference algorithm can use a broad class of existing image registration algorithms, i.e., algorithms for aligning two different images. We interpret our inference algorithm as a way to align a group of images while applying spatially-adaptive intensity equalization per image.

- We demonstrate deformation-invariant sparse coding on neuroimaging data from a language study. Our method identifies activation regions that agree with known literature on language processing and establishes correspondences among activation regions across individuals, producing more robust group-level effects than anatomical alignment alone.

Outline. We provide background material in Chapter 2. Our probabilistic deformation-invariant sparse coding model is presented in Chapter 3 and is used to find functional units in the brain for language processing in Chapter 4. We conclude in Chapter 5.

Chapter 2

Background

We begin this chapter by describing how images and deformations are represented throughout this thesis including notation used. We then provide background material on sparse coding, estimating deformations for aligning images, and finding group-level brain activations evoked by functional stimuli in functional magnetic resonance imaging (fMRI).

■ 2.1 Images, Deformations, Qualitative Spaces, and Masks

To represent images and deformations, we first define the space in which they exist. Consider a finite, discrete set of points $\Omega \subset \mathbb{R}^d$ that consists of coordinates in d-dimensional space that are referred to as *pixels* for 2D images ($d = 2$) and volumetric pixels or *voxels* for 3D images ($d = 3$). For simplicity, we refer to elements of Ω as voxels when working with signals that are not 3D images.

We represent an image in two different ways: as a vector in $\mathbb{R}^{|\Omega|}$ and as a function that maps Ω to $\mathbb{R}$. Specifically, for an image I, we write $I \in \mathbb{R}^{|\Omega|}$ (vector representation) and use indexing notation $I(x) \in \mathbb{R}$ to mean the intensity value of image I at voxel $x \in \Omega$ (functional representation). These two representations are equivalent: by associating each voxel $x \in \Omega$ with a unique index in $\{1, 2, \ldots, |\Omega|\}$, value $I(x)$ becomes just the value of vector $I \in \mathbb{R}^{|\Omega|}$ at the index associated with voxel x.

But what if we want to know the value of an image at a voxel that's not in Ω? To handle this, we extend notation by allowing indexing into an image $I \in \mathbb{R}^{|\Omega|}$ by a voxel that may not be in Ω. Specifically, we allow indexing into a voxel in Ω_c, which is a *continuous extension* Ω_c of Ω, where formally Ω_c is a simply-connected open set that contains Ω. This means that Ω_c is a region comprising of a single connected component, does not have any holes in it, and contains the convex hull of Ω. Then $I(y)$ for $y \in \Omega_c \setminus \Omega$ refers to an interpolated value of image I at voxel $y \notin \Omega$; e.g., nearestneighbor interpolation would simply involve finding $x \in \Omega$ closest in Euclidean distance to voxel y and outputting $I(y) \leftarrow I(x)$.

Next, we discuss deformations, which use interpolation. We define a deformation Φ as a mapping from Ω_c to Ω_c. Note that if Φ only mapped from Ω to Ω, then Φ would just be a permutation, which is insufficient for our purposes. We work with deformations that are diffeomorphisms, which means that they are invertible and both the deformations and their inverses have continuous derivatives of all orders. We let $|\mathbf{J}_\Phi(x)|$ denote the Jacobian determinant of Φ evaluated at voxel x. Crucially, $|\mathbf{J}_\Phi(x)|$ can be interpreted as the volume change ratio for voxel x due to deformation Φ, i.e., $|\mathbf{J}_\Phi(x)|$ partial voxels from the input space of Φ warps to voxel x in the output space of Φ. To see this, consider a compactly supported, continuous function $f : \Omega_c \to \mathbb{R}$. From calculus, we have

$$\int_{\Omega_c} f(\Phi^{-1}(x))dx = \int_{\Omega_c} f(x)|\mathbf{J}_\Phi(x)|dx. \tag{2.1}$$

Observe that voxel x has weight $|\mathbf{J}_\Phi(x)|$ in image f while it has weight 1 in image $f \circ \Phi^{-1}$. Thus, due to applying Φ to $f \circ \Phi^{-1}$ to obtain f, the "volume" at voxel x changes from 1 to $|\mathbf{J}_\Phi(x)|$. This intuition of volume change will be apparent when we discuss averaging deformed images later in this chapter. Also, as eq. (2.1) suggests, for Φ to be invertible, we must have $|\mathbf{J}_\Phi(x)| > 0$ for all $x \in \Omega_c$.

We can interpret deformation Φ as a change of coordinates that may potentially be nonlinear; Φ deforms an input space to an output space and while both input and output spaces are Ω_c, they may have very qualitative meanings! For example, for $\Omega_c = \mathbb{R}_+$ (the positive real line) and $\Phi(x) = \log(x + 1)$, if the input space is in units of millimeters, then the output space, while also being $\mathbb{R}_+$, is in units of log millimeters. Thus, each image is associated with a *qualitative space* (e.g., millimeter space, log-millimeter space, the anatomical space of Alice's brain, the anatomical space of Bob's brain).

With an image I and deformation Φ, we can define deformed image $I \circ \Phi \in \mathbb{R}^{|\Omega|}$ using our functional representation for images:

$$(I \circ \Phi)(x) = I(\Phi(x)) \qquad \text{for } x \in \Omega,$$

where $\Phi(x)$ could be in $\Omega_c \setminus \Omega$, requiring interpolation. Importantly, image $I \circ \Phi$ has the interpretation of image I being deformed by Φ such that $I \circ \Phi$ now has coordinates defined by the input space of Φ.

Henceforth, when dealing with images, we often omit writing out the voxel space Ω and liberally switch between using vector and functional representations for images. We typically use variable x to denote a voxel. In this work, we consider diffeomorphisms and note that by setting $\Omega_c = \mathbb{R}^d$, then translations, rotations, and invertible affine transformations are all examples of diffeomorphisms mapping Ω_c to Ω_c.

■ 2.2 Sparse Coding

As mentioned previously, sparse coding refers to representing an input signal as a sparse linear combination of dictionary elements. For example, sparse coding applied to natural images can learn dictionary elements resembling spatial receptive fields of neurons in the visual cortex [27, 28]. Applied to images, video, and audio, sparse coding can learn dictionary elements that represent localized bases [12, 23, 25, 27, 28, 40]. In this section, we review sparse coding, its associated optimization problem, its probabilistic interpretation, and its relation to factor analysis.

In sparse coding, we model observations $I_1, I_2 \ldots, I_N \in \mathbb{R}^P$ to be generated from dictionary elements $D_1, D_2, \ldots, D_K \in \mathbb{R}^P$ as follows:

$$I_n = \sum_{k=1}^{K} w_{nk} D_k + \varepsilon_n \qquad \text{for } n = 1, 2, \ldots, N, \tag{2.2}$$

where weights $w_n \in \mathbb{R}^K$ are sparse (i.e., mostly zero), and noise $\varepsilon_n \in \mathbb{R}^d$ is associated with observation n. For notational convenience, we write eq. (2.2) in matrix form:

$$\boldsymbol{I} = \boldsymbol{D}\boldsymbol{w} + \boldsymbol{\varepsilon}, \tag{2.3}$$

where we stack column vectors to form matrices $\boldsymbol{I} = [I_1|I_2|\cdots|I_N] \in \mathbb{R}^{P\times N}$, $\boldsymbol{D} = [D_1|D_2|\cdots|D_K] \in \mathbb{R}^{P\times K}$, $\boldsymbol{w} = [w_1|w_2|\cdots|w_N] \in \mathbb{R}^{K\times N}$, and $\boldsymbol{\varepsilon} = [\varepsilon_1|\varepsilon_2|\cdots|\varepsilon_N] \in \mathbb{R}^{P\times N}$. We aim to find dictionary $\boldsymbol{D}$ and sparse weights $\boldsymbol{w}$ that minimize data-fitting error $\|\boldsymbol{I} - \boldsymbol{D}\boldsymbol{w}\|_F^2 = \sum_{n=1}^{N} \|I_n - \boldsymbol{D}w_n\|_2^2$, where $\|\cdot\|_F$ and $\|\cdot\|_2$ refer to the Frobenius and Euclidean norms, respectively.

However, as written, finding dictionary $\boldsymbol{D}$ and sparse weights $\boldsymbol{w}$ is an ill-posed problem because scaling weight w_{nk} by some constant $c > 0$ for all n while scaling dictionary element D_k by $1/c$ results in the same observation $\boldsymbol{I}$. Thus, we require a constraint on either the weights or the dictionary elements. Often a constraint is placed on the latter by requiring $\|D_k\|_2 \le 1$ for each k. A less worrisome issue is that permuting the dictionary elements and their associated weights also yields the same observed signal; this is addressed by just recognizing that the ordering of estimated dictionary elements is not unique.

Actually finding dictionary elements and sparse weights requires that we prescribe a relevant optimization problem. We begin with one such optimization problem:

$$\min_{\boldsymbol{w},\,\boldsymbol{D}} \left\{ \|\boldsymbol{I} - \boldsymbol{D}\boldsymbol{w}\|_F^2 + \lambda \sum_{n=1}^{N} \|w_n\|_1 \right\} \quad \text{subject to:} \quad \|D_k\|_2 \leq 1 \text{ for } k = 1, \dots, K, \tag{2.4}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm, which encourages sparsity [35], and constant $\lambda \geq 0$ trades off minimizing data-fitting error versus sparsity of the weights. Increasing λ favors sparser weights at the expense of possibly increased data-fitting error. Note that we could swap the ℓ_1 norm with a different regularizer provided that it encourages sparsity.

Optimization problem (2.4) is block convex but not jointly convex in weights $\boldsymbol{w}$ and dictionary $\boldsymbol{D}$, so a common strategy for numerical optimization is the following alternating minimization scheme:

1. Hold dictionary $\boldsymbol{D}$ constant and minimize over weights $\boldsymbol{w}$. We can minimize over each w_n separately by solving the following convex optimization problem, referred to as the *Lasso* [35]:

$$\min_{w_n \in \mathbb{R}^K} \|I_n - \boldsymbol{D}w_n\|_2^2 + \lambda \|w_n\|_1 \quad \text{for } n = 1, \dots, N. \tag{2.5}$$

2. Hold weights $\boldsymbol{w}$ constant and minimize over dictionary $\boldsymbol{D}$, which involves solving the following convex optimization problem:

$$\min_{\boldsymbol{D} \in \mathbb{R}^{P \times K}} \|\boldsymbol{I} - \boldsymbol{D}\boldsymbol{w}\|_F^2 \quad \text{subject to:} \quad \|D_k\|_2 \leq 1 \text{ for } k = 1, \dots, K. \tag{2.6}$$

While both steps are convex and can be solved by general purpose convex program solvers, exploiting structure in the sparse coding problem enables more efficient optimization algorithms, such as that of Lee *et al.* [22].

Optimization problem (2.4) has a probabilistic interpretation. Letting each ε_n consist of i.i.d. $\mathcal{N}(0, \sigma^2)$ entries and giving each scalar weight w_{nk} a Laplace prior

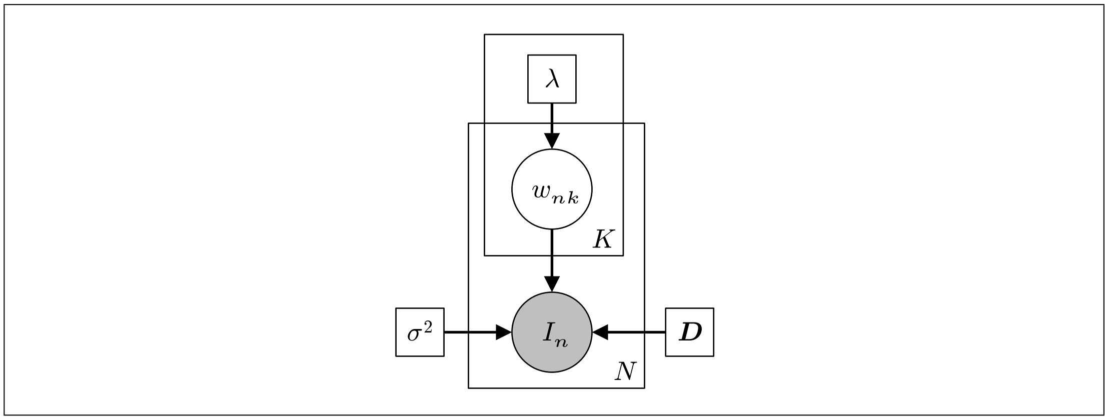


Figure 2.1: A probabilistic graphical model for sparse coding.

$p(w_{nk};\lambda) \propto \exp(-\lambda|w_{nk}|)$, eq. (2.2) implies a probability distribution

$$\begin{aligned} p(\boldsymbol{I},\boldsymbol{w};\boldsymbol{D},\lambda,\sigma^2) &= \prod_{n=1}^{N} p(w_n;\lambda)p(I_n|w_n;\boldsymbol{D},\sigma^2) \\ &\propto \prod_{n=1}^{N} e^{-\lambda\|w_n\|_1}\mathcal{N}(I_n;\boldsymbol{D}w_n,\sigma^2\mathbb{I}_{P\times P}) \\ &\propto \exp\left\{-\lambda\sum_{n=1}^{N}\|w_n\|_1 - \frac{1}{2\sigma^2}\|\boldsymbol{I}-\boldsymbol{D}\boldsymbol{w}\|_F^2\right\}, \end{aligned} \tag{2.7}$$

where $\mathbb{I}_{P\times P}$ is the P-by-P identity matrix, not to be confused with observed images $\boldsymbol{I}$. A graphical model representation is given in Fig. 2.1. Dictionary $\boldsymbol{D}$ and variance σ^2 are treated as parameters, where we constrain $\|D_k\|_2 \le 1$ for each k. However, these variables can also be treated as random with prior distributions. As a preview, our formulation of deformation-invariant sparse coding treats the dictionary $\boldsymbol{D}$ as a random variable and variance σ^2 as a constant.

With $\boldsymbol{I}$ observed, maximizing $p(\boldsymbol{w}|\boldsymbol{I};\boldsymbol{D},\lambda,\sigma^2)$ over $(\boldsymbol{w},\boldsymbol{D})$ is equivalent to minimizing negative $\log p(\boldsymbol{I},\boldsymbol{w};\boldsymbol{D},\lambda,\sigma^2)$ over $(\boldsymbol{w},\boldsymbol{D})$, given by the following optimization problem:

$$\min_{\boldsymbol{w},\,\boldsymbol{D}} \left\{\frac{1}{2\sigma^2}\|\boldsymbol{I}-\boldsymbol{D}\boldsymbol{w}\|_F^2 + \lambda\sum_{n=1}^{N}\|w_n\|_1\right\} \quad \text{subject to:} \quad \|D_k\|_2 \le 1 \text{ for } k=1,\ldots,K. \tag{2.8}$$

This is equivalent to optimization problem (2.4) with λ in (2.4) replaced by $2\lambda\sigma^2$.

We end this section by relating sparse coding to factor analysis. In particular, if the weights were given i.i.d. $\mathcal{N}(0,1)$ priors instead, then we get a factor analysis model, where $\boldsymbol{D}$ is referred to as the *loading matrix* and $\boldsymbol{w}$ consists of the *factors*, which are no longer encouraged to be sparse due to the Gaussian prior. A key feature is that with $\boldsymbol{D}$ fixed, estimating the factors for a signal just involves applying a linear transformation to the signal. Also, the number of factors K per signal is selected to be less than P, the dimensionality of each of I_n, and so factor analysis can be thought of as a linear method for dimensionality reduction whereby we represent $I_n \in \mathbb{R}^P$ using a lower-dimensional representation $w_n \in \mathbb{R}^K$ residing in a subspace of $\mathbb{R}^P$. In contrast, while sparse coding is based on a linear generative model, once we fix the dictionary, estimating weights for an observed signal involves solving the Lasso rather than just applying a linear transformation. Furthermore, sparse coding does not necessarily perform dimensionality reduction since in many applications of sparse coding we have $K > P$. Thus, we can view sparse coding as nonlinearly mapping $I_n \in \mathbb{R}^P$ to $w_n \in \mathbb{R}^K$, achieving dimensionality reduction only if $K < P$.

■ 2.3 Estimating a Deformation that Aligns Two Images

When extending sparse coding to handle deformations, we need to specify what class of deformations we want to consider, e.g., translations, invertible affine transformations, diffeomorphisms. Many such classes already have existing *image registration* algorithms for estimating a deformation that aligns or *registers* two images. For example, we can estimate a diffeomorphism that aligns two images using the diffeomorphic Demons algorithm [36]. In this section we briefly describe how image registration is formulated as an optimization problem and outline Demons registration and its diffeomorphic variant, the latter of which is used when applying deformation-invariant sparse coding to neuroimaging data in Chapter 4.

■ 2.3.1 Pairwise Image Registration as an Optimization Problem

In general, registering a pair of images I and J can be formulated as finding a deformation Φ that minimizes energy

$$E_{\text{pair}}(\Phi; I, J) = \frac{1}{\sigma_i^2}\text{Sim}(I \circ \Phi, J) + \frac{1}{\sigma_T^2}\text{Reg}(\Phi), \tag{2.9}$$

where $\text{Sim}(\cdot,\cdot)$ measures how similar two images are, $\text{Reg}(\cdot)$ measures how complicated a deformation is, and constants $\sigma_i^2, \sigma_T^2 > 0$ trade off how much we favor minimizing the similarity term over minimizing deformation complexity. Image I is said to be the "moving" image since we are applying the deformation Φ to I in the similarity term whereas image J is the "fixed" image. As an example, for estimating a translation, we could have $\text{Sim}(I \circ \Phi, J) = \|I \circ \Phi - J\|_2^2$ and $\text{Reg}(\Phi) = 0$ if Φ is a translation and $\text{Reg}(\Phi) = \infty$ otherwise.

■ 2.3.2 Diffeomorphic Demons Registration

For aligning images of, say, two different people's brains, a simple deformation like a translation is unlikely to produce a good alignment. Instead, we could use a deformation with a "dense" description, specifying where each voxel gets mapped to. Intuitively, we would like to obtain a deformation that is smooth, where adjacent voxels in the moving image aren't mapped to wildly different voxels in the fixed image. Moreover, we would like the deformation to be invertible since if we can warp image I to be close to image J, then we should be able to apply the inverse warp to J to get an image close to I. This motivates seeking a deformation that is a diffeomorphism, which is both smooth and invertible. We now review *log-domain diffeomorphic Demons* [37], which is an algorithm that estimates a diffeomorphism for aligning two images.

The key idea is that we can parameterize a diffeomorphism Φ by a velocity field $\mathcal{V}_\Phi$, where $\Phi = \exp(\mathcal{V}_\Phi)$ and the exponential map for vector fields is defined in [2] and can be efficiently computed via Alg. 2 of [36]. Importantly, the inverse of Φ is given by $\Phi^{-1} = \exp(-\mathcal{V}_\Phi)$. So if we work in the *log domain* defined by the space of velocity fields and exponentiate to recover deformations, then we can rest assured that such resulting deformations are invertible.

Parameterizing diffeomorphisms by velocity fields, log-domain diffeomorphic Demons registration estimates diffeomorphism Φ for aligning moving image I to fixed image J by minimizing energy

$$E_{\text{pair}}(\Phi; I, J) = \min_{\Gamma=\exp(\mathcal{V}_\Gamma)} \left\{ \frac{1}{2\sigma_i^2}\|I \circ \Phi - J\|_2^2 + \frac{1}{\sigma_c^2}\|\log(\Gamma^{-1} \circ \Phi)\|_V^2 + \frac{1}{\sigma_T^2}\text{Reg}(\log(\Gamma)) \right\}$$
$$\text{subject to:} \quad \Phi = \exp(\mathcal{V}_\Phi), \tag{2.10}$$

where Γ is an auxiliary deformation, norm $\|\cdot\|_V$ for a vector field is defined such that $\|u\|_V^2 \triangleq \sum_x \|u(x)\|_2^2$ ($u(x)$ is a velocity vector at voxel x), constants $\sigma_i^2, \sigma_c^2, \sigma_T^2 > 0$

trade off the importance of the three terms, and $\text{Reg}(\cdot)$ is a fluid-like or diffusion-like deformation regularization from [6] that encourages deformation Φ to be smooth albeit indirectly through auxiliary deformation Γ. Essentially $\text{Reg}(\cdot)$ is chosen so that if we fix Φ and minimize over Γ, then the resulting optimization problem just involves computing a convolution. In fact, this is possible [5] provided that $\text{Reg}(\cdot)$ is isotropic and consists of a sum of squared partial derivatives (e.g., $\text{Reg}(\mathcal{V}) = \|\nabla\mathcal{V}\|_V^2$); such a regularization function is referred to as an *isotropic differential quadratic form* (IDQF). Thus, in practice, often $\text{Reg}(\cdot)$ is not specified explicitly and instead Gaussian blurring is used to update auxiliary deformation Γ. Framed in terms of the general pairwise image registration energy (2.9), log-domain diffeomorphic Demons has $\text{Sim}(I \circ \Phi, J) = \frac{1}{2}\|I \circ \Phi - J\|_2^2$ and replaces $\frac{1}{\sigma_T^2}\text{Reg}(\cdot)$ in eq. (2.9) with function

$$\text{LogDiffDemonsReg}(\Phi) = \min_{\Gamma=\exp(\mathcal{V}_\Gamma)} \left\{ \frac{1}{2\sigma_c^2}\|\log(\Gamma^{-1} \circ \Phi)\|_V^2 + \frac{1}{\sigma_T^2}\text{Reg}(\log(\Gamma)) \right\}, \quad (2.11)$$

which still only depends on Φ as σ_c^2 and σ_T^2 are treated as constants.

We sketch the strategy typically used to numerically minimize energy (2.10). For simplicity, we consider the case where $\text{Reg}(\cdot)$ is a diffusion-like regularization, which just means that $\text{Reg}(\cdot)$ is an IDQF as defined in [5].[1] A key idea is that we switch between eq. (2.10) and an alternative form of eq. (2.10) resulting from the following change of variables: Denoting $\Phi = \Gamma \circ \exp(u) = \exp(\mathcal{V}_\Gamma) \circ \exp(u)$, we can rewrite energy (2.10) as

$$E_{\text{pair}}(\Phi; I, J) = \min_{\mathcal{V}_\Gamma} \left\{ \frac{1}{2\sigma_i^2}\|I \circ \exp(\mathcal{V}_\Gamma) \circ \exp(u) - J\|_2^2 + \frac{1}{\sigma_c^2}\|u\|_V^2 + \frac{1}{\sigma_T^2}\text{Reg}(\mathcal{V}_\Gamma) \right\},$$
$$\text{subject to:} \quad \Phi = \exp(\mathcal{V}_\Gamma) \circ \exp(u). \quad (2.12)$$

Let $\hat{\Phi} = \exp(\hat{\mathcal{V}}_\Phi)$ denote the current estimate of $\Phi = \exp(\mathcal{V}_\Phi)$ and $\hat{\Gamma} = \exp(\hat{\mathcal{V}}_\Gamma)$ denote the current estimate of $\Gamma = \exp(\mathcal{V}_\Gamma)$ in the inner optimization problem. After specifying some initial guess for $\hat{\mathcal{V}}_\Gamma$, we minimize (2.10) by iterating between the following two steps:

- With $\hat{\Gamma}$ fixed, minimize energy in form (2.12), which amounts to solving:

$$\hat{u} \leftarrow \underset{u}{\text{argmin}} \left\{ \frac{1}{\sigma_i^2}\|I \circ \exp(\hat{\mathcal{V}}_\Gamma) \circ \exp(u) - J\|_2^2 + \frac{1}{\sigma_c^2}\|u\|_V^2 \right\}. \quad (2.13)$$

[1] A fluid-like regularization function, in addition to being an IDQF, depends on incremental changes, i.e., $\text{Reg}(\mathcal{V}) = f(\mathcal{V}^{(i)} - \mathcal{V}^{(i-1)})$ where i is the iteration number and $f(\cdot)$ is an IDQF.

By modifying the noise variance to be voxel-dependent with estimate $\sigma_i^2(x) = |I \circ \hat{\Phi}(x) - J(x)|^2$ (and thus no longer a known constant) and applying a Taylor approximation, we obtain a closed-form solution [36]:

$$\hat{u}(x) \leftarrow -\left(\frac{J(x) - \hat{I}(x)}{\| - \nabla \hat{I}(x)^T \|_2^2 + \frac{\sigma_i^2(x)}{\sigma_c^2}} \right) \nabla \hat{I}(x)^T, \quad \text{where } \hat{I} \triangleq I \circ \hat{\Gamma}. \tag{2.14}$$

Then update $\hat{\mathcal{V}}_\Phi \leftarrow \log(\exp(\hat{\mathcal{V}}_\Gamma) \circ \exp(\hat{u}))$ using the Baker-Campbell-Hausdorff approximation:

$$\hat{\mathcal{V}}_\Phi \leftarrow \hat{\mathcal{V}}_\Gamma + \hat{u} + \frac{1}{2}[\hat{\mathcal{V}}_\Gamma, \hat{u}], \tag{2.15}$$

where Lie bracket image $[\cdot,\cdot]$ is defined as

$$[\hat{\mathcal{V}}_\Gamma, \hat{u}](x) \triangleq |\mathbf{J}_{\hat{\mathcal{V}}_\Gamma}(x)|\hat{u}(x) - |\mathbf{J}_{\hat{u}}(x)|\hat{\mathcal{V}}_\Gamma(x) \qquad \text{for voxel } x. \tag{2.16}$$

Finally update $\hat{\Phi} \leftarrow \exp(\hat{\mathcal{V}}_\Phi)$.

- With $\hat{\Phi} = \exp(\hat{\mathcal{V}}_\Phi)$ fixed, minimize energy in form (2.10), which amounts to solving:

$$\begin{aligned} \hat{\mathcal{V}}_\Gamma \leftarrow & \operatorname*{argmin}_{\mathcal{V}_\Gamma} \left\{ \frac{1}{\sigma_c^2} \| \log(\Gamma^{-1} \circ \hat{\Phi}) \|_V^2 + \frac{1}{\sigma_T^2} \mathrm{Reg}(\log(\Gamma)) \right\} \\ = & \operatorname*{argmin}_{\mathcal{V}_\Gamma} \left\{ \frac{1}{\sigma_c^2} \| \log(\exp(-\mathcal{V}_\Gamma) \circ \exp(\hat{\mathcal{V}}_\Phi)) \|_V^2 + \frac{1}{\sigma_T^2} \mathrm{Reg}(\mathcal{V}_\Gamma) \right\} \\ \approx & \operatorname*{argmin}_{\mathcal{V}_\Gamma} \left\{ \frac{1}{\sigma_c^2} \| \hat{\mathcal{V}}_\Phi - \mathcal{V}_\Gamma \|_V^2 + \frac{1}{\sigma_T^2} \mathrm{Reg}(\mathcal{V}_\Gamma) \right\}. \end{aligned} \tag{2.17}$$

  As discussed in Section 3 of [5], the solution to the above optimization problem is $\hat{\mathcal{V}}_\Gamma \leftarrow K_{\text{diff}} * \hat{\mathcal{V}}_\Phi$, where "$*$" denotes convolution and K_{diff} is some convolution kernel.

Importantly, introducing auxiliary deformation Γ enables the above alternating minimization with two relatively fast steps. As shown in [6], to handle fluid-like regularization, the only change is that in the first step, after solving optimization (2.13), we immediately set $\hat{u} \leftarrow K_{\text{fluid}} * \hat{u}$ for some convolution kernel K_{fluid}. Typically, convolution kernels K_{diff} and K_{fluid} are chosen to be Gaussian.

■ 2.4 Estimating Deformations that Align a Group of Images

We now present two approaches to aligning a group of images, referred to as *groupwise registration*. One approach is parallelizable across images whereas the other is not. Our inference algorithm for deformation-invariant sparse coding uses the latter as part of initialization and, beyond initialization, can be viewed as an extension to the former.

Both approaches seek invertible, differentiable deformations $\boldsymbol{\Phi} = \{\Phi_1, \Phi_2, \dots, \Phi_N\}$ that align images $\boldsymbol{I} = \{I_1, I_2, \dots, I_N\}$ to obtain average image J, which is "close" to deformed images $I_1 \circ \Phi_1, I_2 \circ \Phi_2, \dots, I_N \circ \Phi_N$. Specifically, they find deformations $\boldsymbol{\Phi}$ and average image J by numerically minimizing energy

$$E_{\text{group}}(\boldsymbol{\Phi}, J; \boldsymbol{I}) = \sum_{n=1}^{N} E_{\text{pair}}(\Phi_n; I_n, J) \tag{2.18}$$

subject to an "average deformation" being identity, which we formalize shortly. In general, without constraining the ensemble of deformations, the problem is ill-posed since modified deformations $\Phi_1 \circ \Gamma, \Phi_2 \circ \Gamma, \dots, \Phi_N \circ \Gamma$ for an invertible deformation Γ could result in the same total energy (e.g., let $E_{\text{pair}}(\Phi; I, J) = \|I \circ \Phi - J\|_2^2 + \|\nabla \Phi\|_V^2$, restrict Φ to be a diffeomorphism, and take Γ to be any translation). The idea is that the qualitative space of average image J could be perturbed without changing the overall energy! Thus, an average deformation constraint anchors the qualitative space of average image J.

We use the following average deformation constraint on diffeomorphisms $\Phi_1 = \exp(\mathcal{V}_1), \dots, \Phi_N = \exp(\mathcal{V}_N)$:

$$\frac{1}{N} \sum_{n=1}^{N} \mathcal{V}_n(x) = 0 \qquad \text{for each voxel } x. \tag{2.19}$$

If the deformations are sufficiently small, then this constraint approximately corresponds to requiring $\Phi_1 \circ \Phi_2 \circ \cdots \circ \Phi_N$ to be identity since $\Phi_1 \circ \Phi_2 \circ \cdots \circ \Phi_N \approx \exp(\sum_{n=1}^{N} \mathcal{V}_n)$.

To modify deformations $\widetilde{\Phi}_1 = \exp(\widetilde{\mathcal{V}}_1), \dots, \widetilde{\Phi}_N = \exp(\widetilde{\mathcal{V}}_N)$ to obtain deformations $\hat{\Phi}_1 = \exp(\hat{\mathcal{V}}_1), \dots, \hat{\Phi}_N = \exp(\hat{\mathcal{V}}_N)$ that satisfy the above constraint, we compute:

$$\hat{\mathcal{V}}_n(x) \leftarrow \widetilde{\mathcal{V}}_n(x) - \frac{1}{N} \sum_{m=1}^{N} \widetilde{\mathcal{V}}_n(x) \qquad \text{for each voxel } x. \tag{2.20}$$

Before plunging into the parallel and serial approaches to groupwise registration, we discuss how to compute average image J if we fix the deformations Φ. This computation depends on the form of the pairwise registration energy E_{pair}. We consider two forms of pairwise energy both based on squared ℓ_2 cost:

- **Average space cost.** The similarity is evaluated in the qualitative space of average image J:

  $$E_{\text{pair}}(\Phi_n; I_n, J) = \|I_n \circ \Phi_n - J\|_2^2 + \text{Reg}(\Phi_n). \tag{2.21}$$

  With deformations $\mathbf{\Phi}$ fixed, minimizing (2.18) with respect to average image J amounts to setting each partial derivative $\partial E_{\text{group}}/\partial J(x)$ to 0 for each voxel x. A straightforward calculation shows that the resulting average image estimate $\hat{J}$ for fixed $\mathbf{\Phi}$ is given by:

  $$\hat{J}(x) \leftarrow \frac{1}{N}\sum_{n=1}^{N} I_n(\Phi_n(x)) \qquad \text{for each voxel } x. \tag{2.22}$$

  More compactly, we can write $\hat{J} \leftarrow \frac{1}{N}\sum_{n=1}^{N} I_n \circ \Phi_n$.

- **Observed space cost.** The similarity is evaluated in the qualitative space of each image I_n:

  $$E_{\text{pair}}(\Phi_n; I_n, J) = \|I_n - J \circ \Phi_n^{-1}\|_2^2 + \text{Reg}(\Phi_n). \tag{2.23}$$

  This form leads to more involved analysis. Letting Ω denote the voxel space and Ω_c a continuous extension of Ω, Eq. (2.1) suggests an approximation:

  $$\sum_{x\in\Omega} f(\Phi_n^{-1}(x)) \approx \int_{\Omega_c} f(\Phi_n^{-1}(x))dx = \int_{\Omega_c} f(x)|\mathbf{J}_{\Phi_n}(x)|dx \approx \sum_{x\in\Omega} |\mathbf{J}_{\Phi_n}(x)| f(x), \tag{2.24}$$

  for which we can take f to be $x \mapsto (I_n(\Phi(x)) - J(x))^2$, and so

  $$E_{\text{pair}}(\Phi_n; I_n, J) \approx \sum_{x} |\mathbf{J}_{\Phi_n}(x)|(I_n(\Phi_n(x)) - J(x))^2 + \text{Reg}(\Phi_n). \tag{2.25}$$

  With this approximation, setting partial derivative $\partial E_{\text{group}}/\partial J(x)$ to 0 for each voxel x yields average image estimate $\hat{J}$ given by:

  $$\hat{J}(x) \leftarrow \frac{\sum_{n=1}^{N} |\mathbf{J}_{\Phi_n}(x)| I_n(\Phi_n(x))}{\sum_{n=1}^{N} |\mathbf{J}_{\Phi_n}(x)|} \qquad \text{for each voxel } x, \tag{2.26}$$

which is a weighted average where the contribution of each $I_n(\Phi(x))$ is weighted by the volume change due to Φ_n at voxel x, e.g., if Φ_n shrinks volume at x, then $|\mathbf{J}_{\Phi_n}(x)|$ is small.

The average space and observed space costs for pairwise image registration differ in that the fixed and moving images are swapped. Note that function $\text{Reg}(\cdot)$ need not be the same for the two different costs and can be chosen so that, in either case, we can apply an existing image registration algorithm. Typically the observed space cost is used in practice because it makes more sense measuring error in the qualitative spaces of observed images, which we can more easily make sense of, rather than in the qualitative space of the average image J, which is essentially a space we're constructing as part of the alignment procedure.

2.4.1 Parallel Groupwise Image Registration

The parallel approach optimizes each pairwise energy in eq. (2.18) independently and then enforces the average deformation constraint before computing the average image. The algorithm proceeds as follows:

Algorithm 1: Parallel Groupwise Image Registration

```
  Input: Images $\boldsymbol{I} = \{I_1, \ldots, I_N\}$
  Output: Aligned image $\hat{J}$, deformations $\hat{\boldsymbol{\Phi}}$ that align input images
1 Make an initial guess $\hat{J}$ for average image $J$, e.g., $\hat{J} \leftarrow \frac{1}{N}\sum_{n=1}^{N} I_n$.
2 repeat
3     for $n = 1, \ldots, N$ do
4         Update $\hat{\Phi}_n$ by solving a pairwise image registration problem
              $\hat{\Phi}_n \leftarrow \underset{\Phi_n}{\text{argmin}}\ E_{\text{pair}}(\Phi_n; I_n, \hat{J})$.
          This step can be parallelized across $n$.
5     Update $\hat{\boldsymbol{\Phi}}$ to satisfy the average deformation constraint using eq. (2.20).
6     Update $\hat{J}$ by computing average image $\hat{J}$ based on images $\boldsymbol{I}$ and estimated
      deformations $\hat{\boldsymbol{\Phi}}$ using eq. (2.22) for the average space cost or eq. (2.26) for
      the observed space cost.
7 until convergence
```

Empirically, initializing $\hat{J}$ with average image $\frac{1}{N}\sum_{n=1}^{N} I_n$ may result in a final average image $\hat{J}$ that is blurry compared to initializing $\hat{J}$ to be one of the images I_n. Moreover, computing the average image after all the deformations have been updated as in line 6 may introduce some blurriness. For example, after doing the first pairwise registration in line 4, if we immediately recomputed the average image $\hat{J}$, then this could affect what estimate $\hat{\Phi}_2$ we obtain and, in practice, can lead to a sharper average image. This leads to the serial approach discussed next.

■ 2.4.2 Serial Groupwise Image Registration

We outline in Alg. 2 a serial groupwise registration approach by Sabuncu *et al.* [31] that essentially recomputes an average image before every pairwise registration and can result in a sharper resulting average image $\hat{J}$. Excluding the current image being registered in line 9 is done to reduce bias in pairwise image registration from line 10 and can be thought of as an implementation detail. Without excluding the current image being registered, lines 7-12 of the algorithm can be viewed as doing coordinate ascent for energy E_{group} simply in a different order than in the parallel approach.

■ 2.5 Finding Group-level Functional Brain Activations in fMRI

To determine what brain regions consistently activate due to a specific task such as language processing or face recognition, we first need some way of measuring brain activity. One way of doing this is to use fMRI, which will be the source of our data in Chapter 4. We provide some fMRI basics before reviewing prior work on finding group-level functional brain activations in fMRI.

FMRI is a widely used imaging modality for observing functional activity in the brain. We specifically consider fMRI that uses the blood-oxygenation-level-dependent (BOLD) magnetic resonance contrast [26]. BOLD fMRI is a non-invasive way to measure the blood oxygenation level in the brain, where local blood flow and local brain metabolism are closely linked [30]. In particular, when neural activity occurs at a certain location in the brain, blood oxygenation level rises around that location, tapering off after the neural activity subsides. As such, fMRI provides an indirect measure of neural activity.

For this thesis, we treat fMRI preprocessing as a black box, referring the reader to prior work [4, 16] for details on standard fMRI preprocessing and to the first three chapters of [21] for an overview of fMRI analysis. The output of the black box, which we treat as the observed fMRI data, consists of one 3D image per individual or *subject*, where each voxel has an intensity value roughly correlated with the voxel being "activated" by a particular stimulus of interest, such as reading sentences. Assuming the stimulus to be the same across a group of individuals, we seek to draw conclusions

Algorithm 2: Serial Groupwise Image Registration

```
Input: Images $\boldsymbol{I} = \{I_1, \ldots, I_N\}$
Output: Aligned image $\hat{J}$, deformations $\hat{\boldsymbol{\Phi}}$ that align input images
/* First pass through data */
begin
    Set initial average image estimate $\hat{J}$ to be one of the images. Without loss of generality, set $\hat{J} \leftarrow I_1$. Set $\hat{\Phi}_1 \leftarrow \mathrm{Id}$.
    for $n = 2, \ldots, N$ do
        Update $\hat{J}$ by computing average image $\hat{J}$ based on images $I_1, I_2, \ldots, I_{n-1}$ deformed by $\hat{\Phi}_1, \hat{\Phi}_2, \ldots, \hat{\Phi}_{n-1}$.
        Update $\hat{\Phi}_n$ by solving a pairwise image registration problem
            $\hat{\Phi}_n \leftarrow \operatorname*{argmin}_{\Phi_n} E_{\mathrm{pair}}(\Phi_n; I_n, \hat{J}).$
    Update $\hat{\boldsymbol{\Phi}}$ to satisfy the average deformation constraint.
/* Subsequent passes through data */
repeat
    for $n = 1, \ldots, N$ do
        Update $\hat{J}$ by computing average image $\hat{J}$ based on images $\boldsymbol{I} \setminus \{I_n\}$ and estimated deformations $\hat{\boldsymbol{\Phi}} \setminus \{\hat{\Phi}_n\}$.
        Update $\hat{\Phi}_n$ by solving a pairwise image registration problem
            $\hat{\Phi}_n \leftarrow \operatorname*{argmin}_{\Phi_n} E_{\mathrm{pair}}(\Phi_n; I_n, \hat{J}).$
    Update $\hat{\boldsymbol{\Phi}}$ to satisfy the average deformation constraint.
until convergence
Update $\hat{J}$ by computing average image $\hat{J}$ based on all images $\boldsymbol{I}$ and all estimated deformations $\hat{\boldsymbol{\Phi}}$.
```

about how the group responds to the stimulus.

As mentioned in the introduction to this thesis, two types of variability pose challenges to accurately assessing the group response:

- **Anatomical variability.** Different subjects' brains are anatomically different. Thus, a popular procedure for obtaining the average group response is to first align all subjects' fMRI data to a common space, such as the Talairach space [32]. This involves aligning each individual's fMRI data to a template brain [7].

- **Functional variability.** Even if we first aligned each subject's brain into a common anatomically-normalized space so that anatomical structures line up perfectly, when given a particular stimulus, different subjects experience brain activations in different locations within the anatomically-normalized space.

After addressing anatomical variability by aligning images to a template brain, the standard approach assumes voxel-wise correspondences across subjects, which means that for a given voxel in the common space, different subjects' data at that voxel are assumed to be from the same location in the brain. Then the group's functional response at a voxel is taken to be essentially the average of the subjects' fMRI data at that voxel. This approach relies on the registration process being perfect and there being no functional variability, neither of which is true [1, 17, 19].

Recent work addresses functional variability in different ways [31, 34, 39]. Thirion *et al.* [34] identify contiguous regions, or *parcels*, of functional activation at the subject level and then find parcel correspondences across subjects. While this approach yields reproducible activation regions and provides spatial correspondences across subjects, its bottom-up, rule-based nature does not incorporate a notion of a group template while finding the correspondences. Instead, it builds a group template as a post-processing step. As such, the model lacks a clear group-level interpretation of the estimated parcels. In contrast, Xu *et al.* [39] use a spatial point process in a hierarchical Bayesian model to describe functional activation regions. Their formulation accounts for variable shape of activation regions and has an intuitive interpretation of group-level activations. However, since the model represents shapes using Gaussian mixture models, functional regions of complex shape could require a large number of Gaussian components. Lastly, Sabuncu *et al.* [31] sidestep finding functional region correspondences altogether by estimating voxel-wise correspondences through groupwise registration of functional activation maps from different subjects. This approach does not explicitly model functional regions.

In this thesis, we propose a novel way to characterize functional variability that combines ideas from [31, 34, 39]. We model each subject's activation map as a weighted sum of group-level functional activation parcels that undergo a subject-specific deformation. Similar to Xu *et al.* [39], we define a hierarchical generative model, but instead of using a Gaussian mixture model to represent shapes, we represent each parcel as an image, which allows for complex shapes. By explicitly modeling parcels, our model yields parcel correspondences across subjects, similar to [34]. Second, we assume that the template regions can deform to account for spatial variability of activation regions across subjects. This involves using groupwise registration similar to [31] that is guided by estimated group-level functional activation regions. We perform inference within the proposed model using an algorithm similar to expectation-maximization (EM) [8] and illustrate our method on the language system, which is known to have significant functional variability [14].

Chapter 3

Probabilistic Deformation-Invariant Sparse Coding

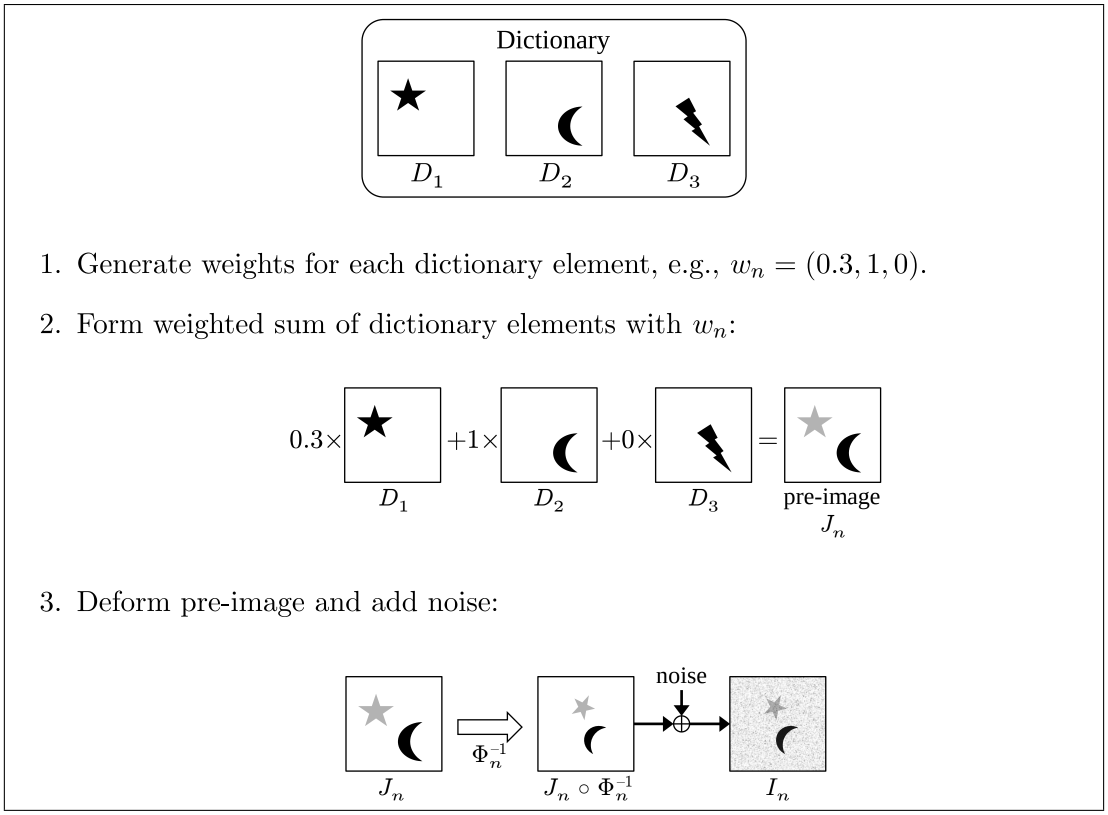
In this chapter, we present our probabilistic model for deformation-invariant sparse coding and provide an accompanying EM-like inference algorithm. For simplicity, we assume signals we deal with to be 3D images defined over uniformly-spaced voxels. Our framework easily extends to other signals defined over uniformly-spaced coordinates.

■ 3.1 Formulation

Let $\boldsymbol{I} = \{I_1, I_2, \ldots, I_N\} \subset \mathbb{R}^{|\Omega|}$ be the N observed images and Ω be a discrete set of voxels our images are defined over; we assume that Ω consists of uniformly-spaced voxels aligned in a grid. While images in $\boldsymbol{I}$ all reside in $\mathbb{R}^{|\Omega|}$, each image has a different qualitative space as discussed in Section 2.1; for example, if I_1 and I_2 are images of two different people's brains, then voxel $x \in \Omega$ in images I_1 and I_2 might not correspond to the same anatomical structure such as the hippocampus because different people's brains vary anatomically. In this example, qualitatively, I_1 lives in subject 1's space and I_2 lives in subject 2's space. Thus, each observation n is associated with a different qualitative space.

Denoting Ω_c to be a continuous extension of Ω, we assume each observation n to have an associated deformation $\Phi_n : \Omega_c \to \Omega_c$ that is diffeomorphic and that these deformations $\boldsymbol{\Phi} = \{\Phi_1, \Phi_2, \ldots, \Phi_N\}$ along with a dictionary of K images $\boldsymbol{D} = \{D_1, D_2, \ldots, D_K\}$ generates the observed images $\boldsymbol{I}$. Dictionary size K is a fixed constant and can be set to the maximum number of dictionary elements we want to consider. As we show later, model parameters can encourage dictionary elements to be 0, so the number of non-zero dictionary elements could be less than K.

We assume that each observed image I_n is generated i.i.d. as follows. First, we



1. Generate weights for each dictionary element, e.g., $w_n = (0.3, 1, 0)$.
2. Form weighted sum of dictionary elements with w_n:

3. Deform pre-image and add noise:

Figure 3.1: Illustration of how our generative model produces observed image I_n for a given dictionary of size $K = 3$.

draw weight vector $w_n \in \mathbb{R}^K$ where each scalar entry w_{nk} is independently sampled from distribution $p_w(\cdot; \lambda_k)$. Then, we construct pre-image $J_n = \sum_{k=1}^K w_{nk} D_k$. The observed image $I_n = J_n \circ \Phi_n^{-1} + \varepsilon_n$ is the result of applying invertible deformation Φ_n^{-1} to pre-image J_n and adding white Gaussian noise ε_n with variance σ^2. This generative process is illustrated in Fig. 3.1 and defines the following joint probability distribution over weight vector w_n and observed image I_n for observation n:

$$p(I_n, w_n | \Phi_n, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) = \left[\prod_{k=1}^{K} p_w(w_{nk}; \lambda_k) \right] \mathcal{N}\left(I_n; \sum_{k=1}^{K} w_{nk} (D_k \circ \Phi_n^{-1}), \sigma^2 \mathbb{I}_{|\Omega| \times |\Omega|} \right), \tag{3.1}$$

where $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \ldots, \lambda_K\}$.

Our goal is to infer dictionary $\boldsymbol{D}$ and deformations $\boldsymbol{\Phi}$ given a training set $\boldsymbol{I}$ of observations so that for future observations that have the same qualitative spaces as our

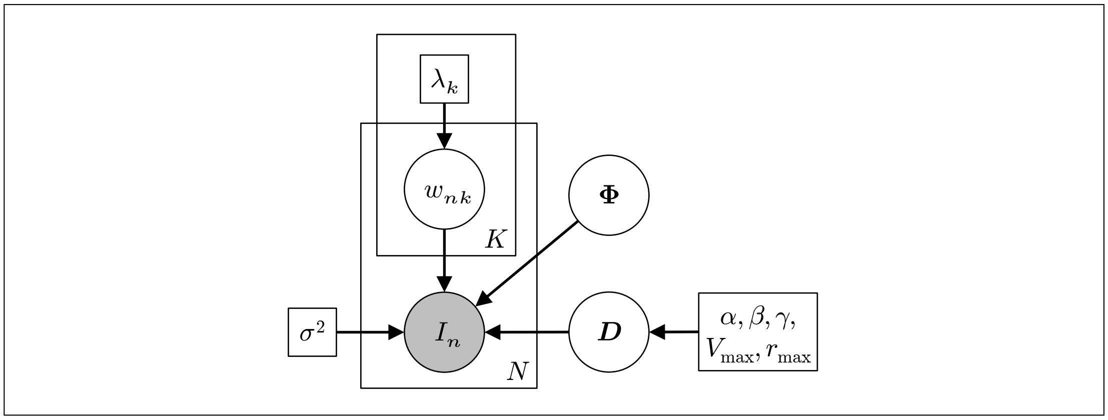


Figure 3.2: A probabilistic graphical model for our generative process.

training observations, we can treat the dictionary and observation-specific deformations as fixed. Thus, the sparse linear combination weights are treated as latent variables during training. Meanwhile, since we don't know $\boldsymbol{\lambda}$ and σ^2, we find maximum-likelihood estimates for these parameters. Mathematically, the resulting inference problem amounts to solving the optimization problem

$$(\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{\lambda}}, \hat{\sigma}^2) \leftarrow \operatorname*{argmin}_{\boldsymbol{D},\boldsymbol{\Phi},\boldsymbol{\lambda},\sigma^2} p(\boldsymbol{\Phi}, \boldsymbol{D} | \boldsymbol{I}; \boldsymbol{\lambda}, \sigma^2). \tag{3.2}$$

For example, in our neuroimaging application, we learn a dictionary and deformations from a training set that comes from N subjects, and we validate on held-out data from the same subjects. At the end of this chapter, we briefly discuss several extensions: (i) estimating sparse linear combination weights $\boldsymbol{w}$ in addition to $(\boldsymbol{D}, \boldsymbol{\Phi}, \boldsymbol{\lambda}, \sigma^2)$, which is an easier problem as the weights are not treated as latent variables; (ii) having deformations be both observation-specific and dictionary-element specific; (iii) handling new observations that have qualitative spaces that aren't in the training observations; and (iv) incorporating ground truth segmentations for supervised learning.

■ 3.1.1 Model Parameters

We treat each deformation Φ_n as a random parameter with prior distribution $p_\Phi(\cdot)$, which can also be viewed as regularizing each deformation to prevent overfitting. Choice of the deformation prior allows us to leverage existing image registration algorithms; specifically, our inference algorithm described in Section 3.2 works with any registration algorithm that minimizes an energy of the form

$$E(\Phi; I, J) = \frac{1}{2\sigma^2}\|I \circ \Phi - J\|_2^2 - \log p_\Phi(\Phi), \tag{3.3}$$

for moving image $I \in \mathbb{R}^{|\Omega|}$ that undergoes deformation Φ and fixed image $J \in \mathbb{R}^{|\Omega|}$, where Φ is restricted to be in a subset of diffeomorphisms mapping Ω_c to Ω_c. To prevent spatial drift of the dictionary elements during inference, we add a constraint that the average deformation be identity where we define the average deformation to be $\Phi_1 \circ \Phi_2 \circ \cdots \circ \Phi_N$. The overall prior on deformations $\mathbf{\Phi}$ is thus

$$p(\mathbf{\Phi}) = \left[\prod_{n=1}^{N} p_\Phi(\Phi_n)\right] \cdot \mathbf{1}\left\{\Phi_1 \circ \cdots \circ \Phi_N = \mathrm{Id}\right\}, \tag{3.4}$$

where $\mathbf{1}\{\cdot\}$ is the indicator function that equals 1 when its argument is true and equals 0 otherwise.

Our inference algorithm depends on the following *volume change condition* on each deformation Φ_n: Letting $|\mathbf{J}_{\Phi_n}(x)|$ denote the Jacobian determinant of Φ_n evaluated at voxel x, we require $|\mathbf{J}_{\Phi_n}(x)| \leq \phi_{\max}$ for all $x \in \Omega$ and for some pre-specified constant $\phi_{\max} \geq 1$. As $|\mathbf{J}_{\Phi_n}(x)|$ is the volume change ratio for voxel x due to deformation Φ_n, the volume change condition can be thought of as a constraint on how much deformation Φ_n is allowed to shrink or expand part of an image (e.g., if Φ_n is identity, then $|\mathbf{J}_{\Phi_n}(x)| = 1$ for all $x \in \Omega$). Rather than folding the volume change condition in as a constraint on each deformation Φ_n, we instead just require that prior p_Φ be chosen so that this condition is satisfied. This condition is essentially a regularity condition, showing up in the derivation for the inference algorithm and also resulting in a rescaling of images when estimating a deformation to align them during inference.

We also treat each dictionary element D_k as a random parameter. Similar to the sparse coding setup in Section 2.2, we resolve the discrepancy between scaling D_k and inversely scaling w_{nk} by constraining each dictionary element D_k to have bounded ℓ_2 norm: $\|D_k\|_2 \leq 1$. To encourage sparsity and smoothness, we introduce ℓ_1 and MRF penalties. To encourage each dictionary element to be a localized basis, we require each D_k to have spatial support (i.e., the set of voxels for which D_k is non-zero) contained within an ellipsoid of pre-specified (maximum) volume $V_{\max}$ and maximum semi-axis length $r_{\max}$; notationally, we denote the set of ellipsoids satisfying these two conditions as $\mathcal{E}(V_{\max}, r_{\max})$.[1] Finally, to discourage overlap between different dictionary elements,

[1] The semi-axis constraint ensures that we don't have oblong ellipsoids that satisfy the volume con-

we place an ℓ_1 penalty on the element-wise product between every pair of distinct dictionary elements. Formally,

$$\begin{aligned} p(\boldsymbol{D}; \alpha, \beta, \gamma, V_{\max}, r_{\max}) \propto \exp\left\{ -\sum_{k=1}^{K} (\alpha \|D_k\|_1 + \frac{\beta}{2} D_k^\top \mathbf{L} D_k) - \gamma \sum_{k \neq \ell} \|D_k \odot D_\ell\|_1 \right\} \\ \cdot \prod_{k=1}^{K} \mathbf{1}\left\{ \begin{array}{c} \|D_k\|_2 \leq 1, \\ \exists \mathcal{E}_k \in \mathcal{E}(V_{\max}, r_{\max}) \text{ s.t. } \mathcal{E}_k \supseteq \text{support}(D_k) \end{array} \right\}, \end{aligned} \tag{3.5}$$

where hyperparameters α, β, γ, $V_{\max}$, and $r_{\max}$ are positive constants, "$\odot$" denotes element-wise multiplication, and $\mathbf{L}$ is the graph Laplacian for a grid graph defined over voxels Ω. As eq. (3.5) suggests, the spatial support of different dictionary elements may be contained by different ellipsoids from $\mathcal{E}(V_{\max}, r_{\max})$.

Other model parameters are treated as non-random: $\boldsymbol{\lambda}$ parameterizes distributions $p_w(\cdot; \lambda_k)$ for each k, and σ^2 is the variance of the Gaussian noise. We use MAP estimation for $\boldsymbol{D}$ and $\boldsymbol{\Phi}$ and ML estimation for $\boldsymbol{\lambda}$ and σ^2. Cross-validation can be used to select hyperparameters α, β, and γ. Hyperparameters $V_{\max}$ and $r_{\max}$ are set to values representative for the maximum spatial support we want our dictionary elements to have.

With the above model parameters, the full joint distribution for our model becomes

$$\begin{aligned} & p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) \\ & \propto p(\boldsymbol{D}) \prod_{n=1}^{N} \left\{ p_\Phi(\Phi_n) \left[\prod_{k=1}^{K} p_w(w_{nk}; \lambda_k) \right] \mathcal{N}\left(I_n; \sum_{k=1}^{K} w_{nk} (D_k \circ \Phi_n^{-1}), \sigma^2 \mathbb{I}_{|\Omega| \times |\Omega|} \right) \right\}, \end{aligned} \tag{3.6}$$

where average deformation $\Phi_1 \circ \cdots \circ \Phi_n$ is identity and $p(\boldsymbol{D})$ refers to eq. (3.5); we omit writing out hyperparameters α, β, γ, $V_{\max}$, and $r_{\max}$ for notational convenience and since we will not be maximizing over these variables with our inference algorithm. A probabilistic graphical model representation for this distribution is given in Fig. 3.2.

straint yet have, for example, one semi-axis length extremely large and all others close to 0. This constraint also allows us to quickly find the spatial support of each dictionary element during inference.

■ 3.1.2 Relation to Sparse Coding

With $\lambda_1 = \cdots = \lambda_K = \lambda$ for some constant $\lambda > 0$, a Laplace prior for p_w, no deformations (i.e., deformations are all identity), and a uniform prior for each D_k on the unit ℓ_2 disk (i.e., $\alpha = \beta = \gamma = 0$ and $V_{\max} = r_{\max} = \infty$), we obtain the probabilistic sparse coding model (2.7) discussed in Section 2.2. We extend sparse coding by allowing dictionary elements to undergo observation-specific deformations. We estimate a set of deformations $\boldsymbol{\Phi}$ and the distribution for latent weights $\boldsymbol{w}$ in addition to learning the dictionary $\boldsymbol{D}$. Effectively, we recover dictionary elements invariant to "small" deformations, where the "size" of a deformation is governed by the deformation prior.

Our deformation-invariant sparse coding model can be interpreted as mapping each observation $I_n \in \mathbb{R}^{|\Omega|}$ to a pair (w_n, Φ_n), where $w_n \in \mathbb{R}^K$ and function Φ_n maps Ω_c to Ω_c. If $K = o(|\Omega|)$ and Φ_n has a sparse description of size $o(|\Omega|)$, then deformation-invariant sparse coding can be interpreted as performing a nonlinear dimensionality reduction. Specifically for deformations, the subset of diffeomorphisms we work with could give Φ_n a sparse description. For example, if Φ_n is an invertible affine transformation, then it is fully characterized by a small matrix. Recent work by Durrleman *et al.* [11] parameterizes diffeomorphisms by a sparse set of control points with accompanying velocity vectors, which can represent more fine-grain deformations than affine transformations.

■ 3.2 Inference

We use an EM-like algorithm[2] to estimate deformations $\boldsymbol{\Phi}$, dictionary $\boldsymbol{D}$, and non-random model parameters $(\boldsymbol{\lambda}, \sigma^2)$. Appendix A contains detailed derivations of the algorithm. To make computation tractable, a key ingredient of the E-step is to approximate posterior distribution $p(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)$ with a fully-factored distribution

$$q(\boldsymbol{w}; \boldsymbol{\psi}) = \prod_{n=1}^{N} \prod_{k=1}^{K} q_w(w_{nk}; \psi_{nk}), \tag{3.7}$$

where distribution $q_w(\cdot; \psi_{nk})$ is parameterized by ψ_{nk}. Importantly, we keep track of the first and second moments of each latent weight w_{nk}, denoted as $\langle \hat{w}_{nk} \rangle \triangleq \mathbb{E}_{\hat{q}_w}[w_{nk}|\boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}]$ and $\langle \hat{w}_{nk}^2 \rangle \triangleq \mathbb{E}_{\hat{q}_w}[w_{nk}^2|\boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}]$, where $\hat{q}_w = q_w(\cdot; \hat{\psi}_{nk})$; collectively the first moments are denoted as $\langle \hat{\boldsymbol{w}} \rangle$ and the second moments as $\langle \hat{\boldsymbol{w}}^2 \rangle$. Effectively the E-step involves computing these moments, which are just expectations. The resulting algorithm is

[2] Due to approximations we make, the algorithm is strictly speaking not EM or even generalized EM.

summarized in Alg. 3.

We elaborate on how each dictionary element estimate $\hat{D}_k$ is updated. By holding all other estimated variables constant, updating $\hat{D}_k$ amounts to numerically minimizing the following energy:

$$\begin{aligned} &E(D_k) \\ &= \frac{1}{2\hat{\sigma}^2}\sum_{n=1}^{N}\sum_{x\in\Omega}|\mathbf{J}_{\hat{\Phi}_n}(x)|\left[\left(I_n(\hat{\Phi}_n(x)) - \sum_{\ell=1}^{K}\langle\hat{w}_{n\ell}\rangle D_\ell(x)\right)^2 + (\langle\hat{w}_{nk}^2\rangle - \langle\hat{w}_{nk}\rangle^2)D_k^2(x)\right] \\ &\quad + \alpha\|D_k\|_1 + \frac{\beta}{2}D_k^\top \mathbf{L} D_k + \gamma\sum_{\ell\neq k}\|D_k\odot D_\ell\|_1, \end{aligned} \tag{3.13}$$

where D_k satisfies $\|D_k\|_2 \leq 1$ and is contained within an ellipsoid of volume $V_{\max}$. This procedure allows for a dictionary element to converge to 0, which would suggest the dictionary element to be extraneous. As for how the numerical minimization is carried out, we first solve a convex relaxation that omits the ellipsoid constraint. The resulting convex problem can be efficiently solved using the fast iterative shrinkage-thresholding algorithm (FISTA) [3], which we specialize for minimizing $E(D_k)$ subject to $\|D_k\|_2 \leq 1$ in Alg. 4. Next, we reintroduce the ellipsoid constraint via the rounding scheme given in Alg. 5. The rounding scheme basically masks the output of the convex program's solution $\widetilde{D}_k$ to an ellipsoid of maximum volume $V_{\max}$ and maximum semi-axis length $r_{\max}$ such that the intensities inside the ellipsoid are large in terms of ℓ_2 norm. A sketch of how this ellipsoid is found:

1. For every ball $\mathcal{B}_c$ of radius $r_{\max}$ with center $c \in \Omega$, compute "mass" image $M(c) = \sum_{x\in\mathcal{B}_c}|\widetilde{D}_k(x)|^2$. Basically $M(c)$ gives us a measure of how much intensity "mass" is preserved by restricting image $\widetilde{D}_k$ to ball $\mathcal{B}_c$.

2. Rank all voxels $c_1, \dots, c_{|\Omega|}$ in Ω so that $M(c_1) \geq M(c_2) \geq \cdots \geq M(c_{|\Omega|})$.

3. For $i = 1, \dots, |\Omega|$: Fit an ellipsoid to voxels in $\mathcal{B}_{c_i}$ that preserves as much ℓ_2 norm as possible in $\widetilde{D}_k$. Keep track of which ellipsoid found so far preserves the most amount of intensity, and break out of the for loop as soon as our best ellipsoid found so far preserves more intensity than any of the remaining balls $\mathcal{B}_{c_{i+1}}, \dots, \mathcal{B}_{c_{|\Omega|}}$.

The first step here can actually be computed efficiently with the help of a fast Fourier transform. To fit an ellipsoid for the third step, we use an approximation that may not yield the best possible ellipsoid within a ball of radius $r_{\max}$.

Algorithm 3: Deformation-Invariant Sparse Coding Inference

```
Input: Observed images $\boldsymbol{I}$, hyperparameters $(\alpha, \beta, \gamma)$
Output: Estimated dictionary $\hat{\boldsymbol{D}}$, deformations $\hat{\boldsymbol{\Phi}}$, model parameters $(\hat{\boldsymbol{\lambda}}, \hat{\sigma}^2)$
Make an initial guess for $\hat{\boldsymbol{D}}$, $\hat{\boldsymbol{\Phi}}$, $\hat{\boldsymbol{\lambda}}$, $\hat{\sigma}^2$, and $\langle \hat{\boldsymbol{w}} \rangle$.
repeat
    /* E-step */
    for $n = 1, \ldots, N$ do
        for $k = 1, \ldots, K$ do
            Update approximating distribution parameter $\hat{\psi}_{nk}$:
                $$\hat{\psi}_{nk} \leftarrow \underset{\psi_{nk}}{\operatorname{argmin}}\ D(q_w(\cdot; \psi_{nk}) \| p_w(\cdot | I_n, \langle \hat{w}_{n \neg k} \rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)), \tag{3.8}$$
            where $D(\cdot \| \cdot)$ denotes Kullback-Leibler divergence, and
            $p_w(\cdot | I_n, \langle \hat{w}_{n \neg k} \rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$ is the posterior distribution of $w_{nk}$ given
            $I_n, \hat{\Phi}_n, \hat{\boldsymbol{D}}$, and $w_{n\ell} = \langle \hat{w}_{n\ell} \rangle$ for $\ell \neq k$.
            Compute expectations $\langle \hat{w}_{nk} \rangle$ and $\langle \hat{w}_{nk}^2 \rangle$ using $\hat{\psi}_{nk}$.
    /* M-step */
    for $n = 1, \ldots, N$ do
        Compute intermediate deformation estimate $\widetilde{\Phi}_n$ by registering rescaled, observed
        image $\sqrt{\phi_{\max}} I_n$ to rescaled, expected pre-image $\sqrt{\phi_{\max}} \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle \hat{D}_k$:
            $$\widetilde{\Phi}_n \leftarrow \min_{\Phi_n} \left\{ \frac{1}{2\sigma^2} \left\| (\sqrt{\phi_{\max}} I_n) \circ \Phi_n - \sqrt{\phi_{\max}} \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle \hat{D}_k \right\|_2^2 - \log p_\Phi(\Phi_n) \right\}. \tag{3.9}$$
        This step can be parallelized across observations.
    for $n = 1, \ldots, N$ do
        Enforce average deformation constraint to update deformation estimate $\hat{\Phi}_n$:
            $$\hat{\Phi}_n \leftarrow \exp\left( \widetilde{\mathcal{V}}_n - \frac{1}{N} \sum_{m=1}^{N} \widetilde{\mathcal{V}}_m \right), \qquad \text{where } \widetilde{\Phi}_n = \exp(\widetilde{\mathcal{V}}_n). \tag{3.10}$$
    for $k = 1, \ldots, K$ do
        Update parameter estimate $\hat{\lambda}_k$:
            $$\hat{\lambda}_k \leftarrow \underset{\lambda_k}{\operatorname{argmax}} \sum_{n=1}^{N} \mathbb{E}_{\hat{q}_w}[\log p_w(w_{nk}; \lambda_k) | \boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}]. \tag{3.11}$$
    Update parameter estimate $\hat{\sigma}^2$:
        $$\hat{\sigma}^2 \leftarrow \frac{1}{N|\Omega|} \sum_{n=1}^{N} \left[ \left\| I_n - \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2 + \sum_{k=1}^{K} (\langle \hat{w}_{nk}^2 \rangle - \langle \hat{w}_{nk} \rangle^2) \| \hat{D}_k \circ \hat{\Phi}_n^{-1} \|_2^2 \right]. \tag{3.12}$$
    for $k$=1,...,$K$ do
        Update $\widetilde{D}_k \leftarrow$ DictionaryElementUpdate($\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{\lambda}}, \hat{\sigma}^2, \alpha, \beta, \gamma, k$) (see Alg. 4).
        Update $\hat{D}_k \leftarrow$ EllipsoidRounding($\widetilde{D}_k, V_{\max}, r_{\max}$) (see Alg. 5).
until convergence
```

Algorithm 4: DictionaryElementUpdate (FISTA)

```
Input: Estimated dictionary $\hat{\boldsymbol{D}}$, deformations $\hat{\boldsymbol{\Phi}}$, model parameters $(\hat{\boldsymbol{\lambda}}, \hat{\sigma}^2)$,
       hyperparameters $(\alpha, \beta, \gamma)$, index $k$ specifiying which dictionary element
       to update
Output: Updated dictionary element $\widetilde{D}_k$
Initialize $t^{(1)} \leftarrow 1$, $\bar{D}_k \leftarrow \widetilde{D}_k^{(1)}$.
Choose step size
    $\delta = \left( \frac{\phi_{\max}}{\hat{\sigma}^2} \sum_{n=1}^{N} \langle \hat{w}_{nk}^2 \rangle + \beta \|\mathbf{L}\|_2 \right)^{-1}, \qquad (3.14)$
where $\|\mathbf{L}\|_2$ is the spectral norm of the grid graph's graph Laplacian $\mathbf{L}$.
Theorem 1.2 of [41] implies $\|\mathbf{L}\|_2 \leq 4d$, where $d$ is the dimensionality of the grid
graph, so it suffices to pick smaller step size $\delta$ by substituting $4d$ in place of $\|\mathbf{L}\|_2$.
Define
$E_{\text{smooth}}(D_k)$
$\triangleq \frac{1}{2\hat{\sigma}^2} \sum_{n=1}^{N} \sum_{x \in \Omega} |\mathbf{J}_{\hat{\Phi}_n}(x)| \left[ \left( I_n(\hat{\Phi}_n(x)) - \sum_{\ell=1}^{K} \langle \hat{w}_{n\ell} \rangle D_\ell(x) \right)^2 + (\langle \hat{w}_{nk}^2 \rangle - \langle \hat{w}_{nk} \rangle^2) D_k^2(x) \right]$
    $+ \frac{\beta}{2} D_k^\top \mathbf{L} D_k. \qquad (3.15)$
Define shrinkage threshold image $T_k$ by
    $T_k(x) = \alpha + \gamma \sum_{\ell \neq k} |\hat{D}_\ell(x)|. \qquad (3.16)$
for $i = 1, 2, \ldots$ until convergence do
    Compute $\widetilde{D}_{k,\text{smooth}}^{(i)} \leftarrow \bar{D}_k - \delta \nabla E_{\text{smooth}}(\bar{D}_k)$,
    where $\nabla E_{\text{smooth}}(\bar{D}_k)(x) = \frac{\partial E_{\text{smooth}}(\bar{D}_k)}{\partial \bar{D}_k(x)}$.
    Apply voxel-dependent shrinkage thresholding:
    $\widetilde{D}_{k,\text{thresholded}}^{(i)} \leftarrow \eta(\widetilde{D}_{k,\text{smooth}}, \delta T_k)$, where
    $\eta(z, \tau)(x) = \text{sign}(z(x)) \max\{|z(x)| - \tau(x), 0\}$.
    Project onto the $\ell_2$ disk:
        $\widetilde{D}_k^{(i)} \leftarrow \begin{cases} \widetilde{D}_{k,\text{thresholded}}^{(i)} & \text{if } \|\widetilde{D}_{k,\text{thresholded}}^{(i)}\|_2 \leq 1, \\ \widetilde{D}_{k,\text{thresholded}}^{(i)} / \|\widetilde{D}_{k,\text{thresholded}}^{(i)}\|_2 & \text{otherwise.} \end{cases}$
    Compute $t^{(i+1)} \leftarrow \frac{1+\sqrt{1+4(t^{(i)})^2}}{2}$.
    Compute $\bar{D}_k \leftarrow \widetilde{D}_k^{(i)} + (\frac{t^{(i)}-1}{t^{(i+1)}})(\widetilde{D}_k^{(i)} - \widetilde{D}_k^{(i-1)})$.
Set $\widetilde{D}_k \leftarrow \widetilde{D}_k^{(i)}$ where $i$ is the final iteration index.
```

Algorithm 5: EllipsoidRounding

```
Input: Image $D$, maximum volume $V_{\max}$, maximum semi-axis length $r_{\max}$
Output: Image $\hat{D}$, which is image $D$ masked to have spatial support contained
        within an ellipsoid of volume $V_{\max}$ and maximum semi-axis length $r_{\max}$
Initialize active set $\Xi \leftarrow \Omega$.
Initialize best ellipsoid found so far $\hat{\mathcal{E}} \leftarrow \emptyset$.
Initialize the preserved squared $\ell_2$ norm of the best ellipsoid found so far to
be $\hat{m} \leftarrow 0$.
Let $B$ be the image associated with a ball of radius $r_{\max}$ centered at the origin:
    $B(x) \triangleq \begin{cases} 1 & \text{if } \|x\|_2 \le r_{\max}, x \in \Omega, \\ 0 & \text{if } \|x\|_2 > r_{\max}, x \in \Omega, \end{cases} \qquad (3.17)$
where without loss of generality, we assume $\Omega$ contains the spatial support of
image $B$.
Compute image $D^2$ consisting of element-wise squared entries of $D$.
Compute intensity "mass" image $M \leftarrow \mathcal{F}^{-1}\{\mathcal{F}\{D^2\} \odot \mathcal{F}\{B\}\}$, where $\mathcal{F}$ denotes
the multi-dimensional discrete Fourier transform, computed via a fast Fourier
transform.
while $\Xi \neq \emptyset$ do
    Let $c \in \Xi$ be a coordinate for which $M(c) \ge M(x)$ for all $x \in \Xi$; this can be
    done quickly if ahead of time we sort voxels by decreasing value of $M(\cdot)$.
    Let $\mathcal{B}_c$ denote the set of voxels that are in the ball of radius $r_{\max}$ centered at
    $c$, and let $Z = \sum_{x \in \mathcal{B}_c} D^2(x)$. Compute
        $v \leftarrow \sum_{x \in \mathcal{B}_c} \frac{D^2(x)}{Z} x, \qquad (3.18)$
        $\mathbf{A}^{-1} \leftarrow \sum_{x \in \mathcal{B}_c} D^2(x)(x - v)(x - v)^\top. \qquad (3.19)$
    Set ellipsoid $\mathcal{E} \leftarrow \{x \in \mathcal{B}_c : (x - v)^\top \mathbf{A}(x - v) \le \xi\}$, where constant $\xi$ ensures
    that $\mathcal{E}$ has volume $V_{\max}$. This fit may not guarantee finding the ellipsoid
    contained within $\mathcal{B}_c$ that preserves the most $\ell_2$ norm of $D$ and is thus an
    approximation.
    Compute $m \leftarrow \sum_{x \in \mathcal{E}} D^2(x)$.
    if $m > \hat{m}$ then Set $\hat{m} \leftarrow m$ and $\hat{\mathcal{E}} \leftarrow \mathcal{E}$.
    Remove voxel $c$ from $\Xi$. Remove every voxel $x$ of active set $\Xi$ where
    $M(x) < m$.
Compute
    $\hat{D}(x) = \begin{cases} D(x) & \text{if } x \in \hat{\mathcal{E}}, \\ 0 & \text{otherwise.} \end{cases} \qquad (3.20)$
```

3.2.1 Initialization

We now detail how line 1 of Alg. 3 is carried out. To initialize deformation estimates $\hat{\boldsymbol{\Phi}}$, we align all the observed images together via the serial groupwise image registration as discussed in Section 2.4. From groupwise registration, we also obtain an "average" image across all observed images. We cluster this average image into K initial dictionary elements $\hat{\boldsymbol{D}} = \{\hat{D}_1, \dots, \hat{D}_K\}$. Different clustering methods can be used. We use watershed segmentation and retain the largest K segments; parameters for watershed segmentation depend on the kind of images that are being used.

Rather than initialize approximating distribution parameters $\hat{\boldsymbol{\psi}}$ for the latent weights, we directly compute guesses for the expected latent weights $\langle \hat{w}_n \rangle \triangleq (\langle \hat{w}_{n1} \rangle, \dots, \langle \hat{w}_{nK} \rangle) \in \mathbb{R}^K$ by solving a least-squares regression problem for each observation n:

$$\langle \hat{w}_n \rangle \leftarrow \underset{w_n \in \mathbb{R}^K}{\operatorname{argmin}} \left\| I_n - \sum_{k=1}^{K} w_{nk} (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2 , \tag{3.21}$$

where we may have to project $\langle \hat{w}_{nk} \rangle$ onto the support of distribution w_{nk}. For example, if w_{nk} is a non-negative random variable and $\langle \hat{w}_{nk} \rangle$ is estimated by the above least-squares optimization to be negative, then we just set $\langle \hat{w}_{nk} \rangle$ to be 0.

Lastly, we compute initial estimates for $\boldsymbol{\lambda}$ and σ^2. We use update eq. (3.11) to get an initial estimate for $\boldsymbol{\lambda}$. As for σ^2, we use the initial estimate of

$$\hat{\sigma}^2 = \frac{1}{N|\Omega|} \sum_{n=1}^{N} \left\| I_n - \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2 . \tag{3.22}$$

3.2.2 Intensity-equalization Interpretation

We can interpret the inference algorithm as a modification of the parallel groupwise image registration algorithm presented in Section 2.4.1 where we now apply spatially-adaptive intensity equalization per image. Specifically, the inference algorithm can be phrased as follows:

1. For each n, estimate expected pre-images $\hat{J}_n \leftarrow \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle \hat{D}_k$.

2. For each n, update $\hat{\Phi}_n$ by solving a pairwise image registration problem

$$\hat{\Phi}_n \leftarrow \underset{\Phi_n}{\operatorname{argmin}}\, E_{\text{pair}}(\Phi_n; I_n, \hat{J}_n),$$

which can be done in parallel across $n = 1, 2, \ldots, N$.

3. Update $\hat{\boldsymbol{\Phi}}$ to satisfy the average deformation constraint.

4. Update the estimates for $\hat{\boldsymbol{D}}$, $\hat{\boldsymbol{\lambda}}$, and $\hat{\sigma}^2$.

The expected pre-image for observation n allows each dictionary element to have a different weight, which means that we adjust the intensity of the expected pre-image by different amounts where different dictionary elements appear.

■ 3.3 Extensions

We remark on a few possible extensions of our model.

- *Estimating sparse linear combination weights instead of treating them as latent*: Regular sparse coding actually does not treat the weights as latent. By estimating weights instead, our inference algorithm would actually become simpler since no variational approximating distribution q over latent weights is needed. For example, if the prior on weights p_w is a Laplace distribution with the same parameter λ across all dictionary elements, then the E-step becomes an instance of Lasso:

$$\min_{w_n \in \mathbb{R}^K} \frac{1}{2\sigma^2} \left\| I_n - \sum_{k=1}^{K} w_{nk}(D_k \circ \Phi_n^{-1}) \right\|_2^2 + \lambda \|w_n\|_1. \tag{3.23}$$

  Moreover, the volume change condition can actually be dropped since it was introduced to determine which image is treated as fixed and which is treated as moving when updating each deformation. Specifically, as shown in eq. (A.5) in Appendix A, there is a term involving $\|D_k \circ \Phi_n^{-1}\|_2^2$ that is present due to the weights being latent and that would require modifying off-the-shelf image registration algorithms to account for it. Introducing the volume change condition and making the expected pre-image the fixed image instead of the moving image allows us to essentially get rid of this term when updating each deformation. However, with weights that are not latent, we could treat the expected pre-image as the moving image.

- *Having deformations that are both observation-specific and dictionary-element-specific*: Our deformations are observation-specific, and depending on the subset of diffeomorphisms we allow, observation-specific deformations could already provide enough flexibility to deform different dictionary elements differently. Of course, we could extend our model to have deformations that are both observation-specific and dictionary-element specific, where the main change will be that in the M-step, we need to perform NK pairwise image registrations to align each observation with K different dictionary elements.

- *Handling new observations with qualitative spaces not seen during training*: This extension is challenging because ideally the deformations should be treated as latent variables as well, but integration over the space of deformations is, in general, non-trivial, especially for deformations as rich as diffeomorphisms. Risholm *et al.* [29] provide a Markov-chain Monte Carlo approach to sampling deformations, which can be used to numerically approximate an integral over deformations. However, this procedure, if used as a subroutine for our inference algorithm (by estimating expected deformations in the E-step and no longer estimating deformations in the M-step), would make the inference algorithm prohibitively slow.

  A far simpler approach would be to use our inference algorithm that on a training dataset and then for a new observation, estimate both the deformation that aligns the qualitative space of the new observation to the qualitative space of the dictionary elements and the contribution of each dictionary element. Specifically, treating $(\boldsymbol{D}, \boldsymbol{\lambda}, \sigma^2)$ as fixed constants learned from training data via our inference algorithm, for new observation I', we would solve

$$(\Phi', w') \leftarrow \underset{\Phi:\Omega_c \to \Omega_c,\, w \in \mathbb{R}^K}{\operatorname{argmax}} \; p(\Phi, w | \boldsymbol{D}, I, \boldsymbol{\lambda}, \sigma^2). \tag{3.24}$$

  This problem can be numerically optimized via an alternating maximization scheme, fixing the deformation and updating the weights then fixing the weights and updating the deformation.

- *Incorporating ground truth segmentations for supervised learning*: In some applications, we could have ground truth segmentations indicating where dictionary elements are in each observed image to be used for learning a dictionary and deformations via our model. Thus, we would like these ground truth segmentations to guide our inference algorithm and effectively place additional soft or hard constraint on the dictionary elements. For example, if each image is of a handwritten digit, then we could have a segmentation specifying where the handwritten digit is and what number it is. Then we would like our dictionary elements learned to correspond to different digits. We now outline a simple way to incorporate ground truth segmentations where we assume that for each image segmentation, no voxel is assigned to more than one dictionary element. The end result will be an inference algorithm that is nearly identical to the one in Section 3.2 where the main change is that we now require an image registration algorithm that aligns two K-channel images, i.e., voxels are assigned a value in $\mathbb{R}^K$.

First, we assume that the number of dictionary elements K is known in this supervised learning setup. For each training image I_n, we assume we have a segmentation of the image into non-overlapping regions, each region assigned to either none of the dictionary elements or exactly one of the dictionary elements. Thus, the segmentation for image I_n can be represented as $(\Omega_{n0}, \Omega_{n1}, \ldots, \Omega_{nK})$, where Ω_{n0} is the set of voxels not assigned to any dictionary element, and Ω_{nk} for $k = 1, 2, \ldots, K$ is the set of voxels assigned to dictionary element k. We have $\bigcup_{k=0}^{K} \Omega_{nk} = \Omega$, the whole voxel space. So we observe $\boldsymbol{I} = \{I_1, \ldots, I_N\}$ where for each I_n we also observe corresponding segmentation $(\Omega_{n0}, \Omega_{n1}, \ldots, \Omega_{nK})$. We refer to the set of all such observed segmentations as $\boldsymbol{\Omega}$.

Then we can employ the following representation:

$$I_n(x) = \sum_{k=0}^{K} S_{nk}(x), \tag{3.25}$$

where

$$S_{nk}(x) \triangleq \begin{cases} w_{nk} D_k(\Phi_n^{-1}(x)) + \varepsilon_n(x) & \text{if } x \in \Omega_{nk}, \\ 0 & \text{otherwise,} \end{cases} \tag{3.26}$$

and by convention $w_{n0} \triangleq 0$. Observing $\boldsymbol{I}$ and $\boldsymbol{\Omega}$ means that we observe segmented images S_{nk} for all n and k. We let $\boldsymbol{S}$ denote the set of all S_{nk} images. Importantly, knowing $\boldsymbol{S}$ means that we can reconstruct all the observed images $\boldsymbol{I}$ as well as the segmentations $\boldsymbol{\Omega}$ except that there is a (benign) ambiguity where voxels that have 0 intensity could be assigned to any dictionary element; we address this ambiguity by declaring such 0-intensity voxels to not be assigned to any dictionary element. We now consider a probabilistic model that treats segmented images $\boldsymbol{S}$ as observations:

$$p(\boldsymbol{S}, \boldsymbol{w} | \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) \\ = \prod_{n=1}^{N} \left\{ \left[\prod_{k=1}^{K} p(w_{nk}; \lambda_k) \right] \left[\prod_{k=1}^{K} \prod_{x \in \Omega_k} \mathcal{N}(S_{nk}(x); w_{nk} D_k(\Phi_n^{-1}(x)), \sigma^2) \right] \right\}. \quad (3.27)$$

We can place priors on sparse linear combination weights, deformations, and the dictionary as in Section 3.1. Treating weights $\boldsymbol{w}$ as latent and inferring deformations $\boldsymbol{\Phi}$, dictionary $\boldsymbol{D}$, and parameters $(\boldsymbol{\lambda}, \sigma^2)$ can be done with an algorithm similar the one in Section 3.2. The main change is that to update estimate $\hat{\Phi}_n$, we align K-channel image $\{S_{nk}\}_{k=1}^{K}$ with K-channel image $\{\langle \hat{w}_{nk} \rangle \hat{D}_k\}_{k=1}^{K}$. If we know that each image has exactly one dictionary element present (e.g., each image is of a handwritten digit), then we just need single-channel image registration.

Specifically for handwritten digits, the inference algorithm would just align all the images of 1's to obtain a dictionary element or template for digit 1 and then repeat this for all the other digits. Then for a new observed image, we estimate the contribution of each dictionary element, where one way to classify the observation is to declare the most likely digit to be the dictionary element with the highest contribution, which essentially amounts to aligning the observed image to each dictionary element and deciding which dictionary element provides the best alignment. Such deformable template models are not new and have been used for handwritten digit and face recognition [18, 24].

A general, probabilistic framework for deformable template models for object recognition is presented in [15], which relies on choosing a dictionary of parts that make sense for the recognition task (e.g., having the parts be eyes, a nose, etc. for face recognition). In contrast, our model is presented in the context of unsupervised learning.

Chapter 4

Modeling Spatial Variability of Functional Patterns in the Brain

We seek group-level functional units (i.e., parcels) in the brain that activate due to language processing by representing them as dictionary elements with deformation-invariant sparse coding. After instantiating our inference algorithm to a specific choice of priors for sparse linear combination weights and deformations, this chapter showcases results on synthetic data and fMRI activation maps from a language fMRI study.

■ 4.1 Instantiation to fMRI Analysis

We specialize our model in Chapter 3 for fMRI analysis. First and foremost, observations directly correspond to different subjects; image I_n gives us a brain activation map for subject n. Since we look for positive activation due to functional stimuli probing lexical and structural language processing, it suffices to restrict weights $\boldsymbol{w}$ in our model to be non-negative. We place i.i.d. exponential priors on each w_{nk} to encourage sparsity: $p_w(w_{nk};\lambda_k) = \lambda_k e^{-\lambda_k w_{nk}}$ where $w_{nk} \geq 0$ and $\lambda_k > 0$. We use log-domain diffeomorphic Demons registration [37] to estimate deformations $\boldsymbol{\Phi}$: $p_\Phi(\Phi_n) \propto \exp\{-\text{LogDiffDemonsReg}(\Phi_n)\}$, where $\text{LogDiffDemonsReg}(\cdot)$ is given by eq. (2.11). Combining these priors for weights $\boldsymbol{w}$ and deformations $\boldsymbol{\Phi}$ with eqs. (3.1), (3.4), and (3.5) yields the full joint distribution:

$$\begin{aligned} &p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) \\ &\propto p(\boldsymbol{D}) \prod_{n=1}^{N} \Bigg\{ \exp(-\text{LogDiffDemonsReg}(\Phi_n)) \\ &\qquad\qquad \left[\prod_{k=1}^{K} \lambda_k e^{-\lambda_k w_{nk}}\right] \mathcal{N}\left(I_n; \sum_{k=1}^{K} w_{nk}(D_k \circ \Phi_n^{-1}), \sigma^2 \mathbb{I}_{|\Omega|\times|\Omega|}\right)\Bigg\}, \end{aligned} \tag{4.1}$$

where weights $\boldsymbol{w}$ are non-negative, the average deformation $\Phi_1 \circ \cdots \circ \Phi_n$ is identity, and $p(\boldsymbol{D})$ refers to eq. (3.5). As before, prior $p(\boldsymbol{D})$ depends on hyperparameters α, β, γ, $V_{\max}$, and $r_{\max}$, which we suppress for notational convenience. While we don't place a hard constraint that each dictionary element D_k be a parcel, i.e., have spatial support that is a contiguous region, we empirically find our ellipsoid constraint to produce dictionary elements that are parcels provided that α, $V_{\max}$, and $r_{\max}$ are not too large.

Meanwhile, despite treating log-domain diffeomorphic Demons registration as a black box, we acknowledge the importance of choosing registration parameters to prevent data overfitting. These parameters largely depend on the images used for registration. In our experiments, we use default parameters from the ITK implementation of log-domain diffeomorphic Demons registration [10] except that we increase the variance of the isotropic Gaussian kernel used for smoothing the deformation's velocity field from (1.5 voxels)^2 to (2.5 voxels)^2 and decrease the maximum length of the velocity update vector from 2 voxels to 1 voxel.

Given our choice of priors on weights and deformations, we can now extract explicit update rules for optimization problems (3.8) and (3.11) in the EM-like algorithm of Chapter 3 while using log-domain Demons registration to solve (3.9). Approximating posterior probability distribution q for latent weights $\boldsymbol{w}$ is set to

$$q(\boldsymbol{w}; \boldsymbol{\mu}, \boldsymbol{\nu}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \mathcal{N}^{+}(w_{nk}; \mu_{nk}, \nu_{nk}), \tag{4.2}$$

where $\mathcal{N}^{+}(\cdot; \mu_{nk}, \nu_{nk})$ is the probability density of the normal distribution with mean μ_{nk} and variance ν_{nk} restricted to have non-negative support, i.e., the positive normal distribution. Thus, ψ_{nk} from eq. (3.7) is given by $\psi_{nk} = (\mu_{nk}, \nu_{nk})$. Deferring derivations to Appendix B, we summarize the resulting inference algorithm in Alg. 6, where we denote $\langle \hat{w}_{nk} \rangle \triangleq \mathbb{E}_{\hat{q}}[w_{nk} | \boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}]$, $\langle \hat{w}_{nk}^2 \rangle \triangleq \mathbb{E}_{\hat{q}}[w_{nk}^2 | \boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}]$, and $\hat{q}$ to be distribution (4.2) parameterized by $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\nu}}$. The initialization procedure expands on that of Section 3.2.1 and is outlined in Alg. 7. Before running the inference algorithm, we must set the hyperparameters. As mentioned previously, hyperparameters $V_{\max}$ and $r_{\max}$ have intuitive interpretations and are hand-set based on the maximum spatial support size we would like each dictionary element to have. Thus, our discussion of hyperparameter tuning will focus on choosing α, β, and γ.

Algorithm 6: Deformation-Invariant Sparse Coding Inference for fMRI Analysis

```
Input: Observed images $\boldsymbol{I}$, hyperparameters $(\alpha, \beta, \gamma)$
Output: Estimated dictionary $\hat{\boldsymbol{D}}$, deformations $\hat{\boldsymbol{\Phi}}$, model parameters $(\hat{\boldsymbol{\lambda}}, \hat{\sigma}^2)$
Make an initial guess for $\hat{\boldsymbol{D}}$, $\hat{\boldsymbol{\Phi}}$, $\hat{\boldsymbol{\lambda}}$, $\hat{\sigma}^2$, and $\langle \hat{\boldsymbol{w}} \rangle$ using Alg. 7.
repeat
    /* E-step */
    for $n = 1, \ldots, N$ do
        for $k = 1, \ldots, K$ do
            Update approximating distribution parameters $\hat{\mu}_{nk}$ and $\hat{\nu}_{nk}$:
                $\hat{\mu}_{nk} \leftarrow \frac{\langle I_n - \sum_{\ell \neq k} \langle \hat{w}_{n\ell} \rangle (\hat{D}_\ell \circ \hat{\Phi}_n^{-1}), \hat{D}_k \circ \hat{\Phi}_n^{-1} \rangle - \hat{\sigma}^2 \hat{\lambda}_k}{\|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2}, \qquad (4.3)$
                $\hat{\nu}_{nk} \leftarrow \frac{\hat{\sigma}^2}{\|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2}, \qquad (4.4)$
            where $\langle \cdot, \cdot \rangle$ denotes the standard inner product.
            Compute expectations $\langle \hat{w}_{nk} \rangle$ and $\langle \hat{w}_{nk}^2 \rangle$:
                $\langle \hat{w}_{nk} \rangle \leftarrow \hat{\mu}_{nk} + \frac{\sqrt{\hat{\nu}_{nk}} \exp(-\hat{\mu}_{nk}^2/(2\hat{\nu}_{nk}))}{\sqrt{2\pi} Q(-\hat{\mu}_{nk}/\sqrt{\hat{\nu}_{nk}})}, \qquad (4.5)$
                $\langle \hat{w}_{nk}^2 \rangle \leftarrow \hat{\nu}_{nk} + \hat{\mu}_{nk}^2 + \frac{\hat{\mu}_{nk} \sqrt{\hat{\nu}_{nk}} \exp(-\hat{\mu}_{nk}^2/(2\hat{\nu}_{nk}))}{\sqrt{2\pi} Q(-\hat{\mu}_{nk}/\sqrt{\hat{\nu}_{nk}})}, \qquad (4.6)$
            where $Q(s) \triangleq \int_s^\infty \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$ is the tail probability of the standard normal distribution.
    /* M-step */
    for $n = 1, \ldots, N$ do
        Compute intermediate deformation estimate $\widetilde{\Phi}_n$ by registering rescaled, observed image $\sqrt{\phi_{\max}} I_n$ to rescaled, expected pre-image $\sqrt{\phi_{\max}} \sum_{k=1}^K \langle \hat{w}_{nk} \rangle \hat{D}_k$ using log-domain diffeomorphic Demons registration; this can be parallelized across $n$.
    for $n = 1, \ldots, N$ do
        Enforce average deformation constraint to update deformation estimate $\hat{\Phi}_n$:
            $\hat{\Phi}_n \leftarrow \exp\left( \widetilde{\mathcal{V}}_n - \frac{1}{N} \sum_{m=1}^N \widetilde{\mathcal{V}}_m \right), \qquad \text{where } \widetilde{\Phi}_n = \exp(\widetilde{\mathcal{V}}_n). \qquad (4.7)$
    for $k = 1, \ldots, K$ do
        Update parameter estimate $\hat{\lambda}_k$: $\hat{\lambda}_k \leftarrow 1/(\frac{1}{N} \sum_{n=1}^N \langle \hat{w}_{nk} \rangle)$.
    Update parameter estimate $\hat{\sigma}^2$:
    $\hat{\sigma}^2 \leftarrow \frac{1}{N|\Omega|} \sum_{n=1}^N \left[ \left\| I_n - \sum_{k=1}^K \langle \hat{w}_{nk} \rangle (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2 + \sum_{k=1}^K (\langle \hat{w}_{nk}^2 \rangle - \langle \hat{w}_{nk} \rangle^2) \|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2 \right]. \qquad (4.8)$
    for k=1,...,$K$ do
        Update $\widetilde{D}_k \leftarrow \text{DictionaryElementUpdate}(\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{\lambda}}, \hat{\sigma}^2, \alpha, \beta, \gamma, k)$ (see Alg. 4).
        Update $\hat{D}_k \leftarrow \text{EllipsoidRounding}(\widetilde{D}_k, V_{\max}, r_{\max})$ (see Alg. 5).
until convergence
```

Algorithm 7: Initialization for fMRI Analysis

```
Input: Observed images $\boldsymbol{I}$
Output: Guesses for estimated dictionary $\hat{\boldsymbol{D}}$, deformations $\hat{\boldsymbol{\Phi}}$, model
        parameters $(\hat{\boldsymbol{\lambda}}, \hat{\sigma}^2)$, and expectations $\langle \hat{\boldsymbol{w}} \rangle$
/* Initialize deformations                                                   */
Compute $\hat{A}, \hat{\boldsymbol{\Phi}}$ via serial groupwise image registration of images $\boldsymbol{I}$ (see Alg. 2).
/* Initialize dictionary using watershed segmentation clustering              */
Choose an intensity threshold $\tau$: For synthetic data, set $\tau \leftarrow 0$. For real fMRI
  data, set $\tau \leftarrow 75^{\text{th}}$ percentile value of $\{\hat{A}(x) : x \in \Omega, \hat{A}(x) > 0\}$.
Compute $\widetilde{A}$ to be a Gaussian-blurred version of $\hat{A}$. For synthetic data, the
  Gaussian blur standard deviation is set to 3 voxels. For real data, we use an
  8mm-FWHM blur.
Compute $S_1, S_2, \ldots, S_K \subset \Omega$, which are the largest $K$ segments (in volume) of
  image $\widetilde{A}$ using watershed segmentation, where only voxels with intensity value
  greater than $\tau$ are considered.
for $k = 1, \ldots, K$ do
    Set
        $$\hat{D}_k \leftarrow \begin{cases} \hat{A}(x) & \text{if } x \in S_k, \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$
/* Initialize expected weights                                                */
for $n = 1, \ldots, N$ do
    Solve:
        $$\langle \hat{w}_n \rangle \leftarrow \underset{w_n \in \mathbb{R}^K}{\operatorname{argmin}} \left\| I_n - \sum_{k=1}^{K} w_{nk} (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2, \tag{4.10}$$
    for $k = 1, \ldots, K$ do
        Set $\langle \hat{w}_{nk} \rangle \leftarrow \max\{\langle \hat{w}_{nk} \rangle, 0\}$.
/* Initialize parameter estimate $\hat{\boldsymbol{\lambda}}$                   */
for $k = 1, \ldots, K$ do Set $\hat{\lambda}_k \leftarrow 1/(\frac{1}{N} \sum_{n=1}^{N} \langle \hat{w}_{nk} \rangle)$.
/* Initialize parameter estimate $\hat{\sigma}^2$                              */
Compute
    $$\hat{\sigma}^2 = \frac{1}{N|\Omega|} \sum_{n=1}^{N} \left\| I_n - \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle (\hat{D}_k \circ \hat{\Phi}_n^{-1}) \right\|_2^2. \tag{4.11}$$
```

4.2 Hyperparameter Selection

Before venturing further, we elaborate on some nuances in hyperparameter tuning. First, setting these hyperparameters as to maximize the likelihood of our model is infeasible because we cannot tractably compute the partition function for the dictionary prior. Second, since dictionary $\boldsymbol{D}$ and deformations $\boldsymbol{\Phi}$ serve as ground truth for both training and test images, we can't use them during training since we'd contaminate training with ground truth from test data. We could instead train on a separate set of observations that have different underlying dictionary elements and deformations. However, for real fMRI data, this would require a training set of subjects that is disjoint from the test set of subjects, which could be problematic if the total number of subjects for a study is small. Also, ultimately we would like to draw conclusions about all subjects—not just subjects used for testing.

In the next section, we select hyperparameters α, β, and γ via cross-validation with limited ground truth. Then in Section 4.2.2, we discuss heuristics for selecting hyperparameters in the absence of ground truth; these heuristics play a pivotal role when we work with real fMRI data. In both cases, we assume that each subject n has three images $I_n^{(0)}, I_n^{(1)}, I_n^{(2)}$ that share the same qualitative space since they're from the same brain. Thus, our data can be partitioned into three sets $\boldsymbol{I}^{(0)} = \{I_1^{(0)}, \dots, I_N^{(0)}\}$, $\boldsymbol{I}^{(1)} = \{I_1^{(1)}, \dots, I_N^{(1)}\}$, and $\boldsymbol{I}^{(2)} = \{I_1^{(2)}, \dots, I_N^{(2)}\}$. Sets $\boldsymbol{I}^{(1)}$ and $\boldsymbol{I}^{(2)}$ are used for training and are called *folds*, where fold 1 contains images $\boldsymbol{I}^{(1)}$ and fold 2 contains images $\boldsymbol{I}^{(2)}$. Set $\boldsymbol{I}^{(0)}$ denotes the test images. For simplicity, we choose hyperparameters based on 2-fold cross-validation, but the selection methods we describe can be generalized to handle more folds.

Notationally, we use $(\hat{\boldsymbol{D}}^{(m)}[\alpha, \beta, \gamma], \hat{\boldsymbol{\Phi}}^{(m)}[\alpha, \beta, \gamma])$ to denote the estimated dictionary and deformations learned from observed images $\boldsymbol{I}^{(m)}$ in fold m using hyperparameters (α, β, γ). We use $(\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(m)}[\alpha, \beta, \gamma]$ to denote dictionary element $\hat{D}_k^{(m)}[\alpha, \beta, \gamma]$ deformed into the space of subject n.

4.2.1 Cross-validation Based on Limited Ground Truth

We now discuss the cost function placed on hyperparameters α, β, and γ. To avoid touching the ground truth dictionary and deformations shared across training and test data, we assume that the only ground truth data we have access to during training are the true warped pre-images $W_n^{(m)} \triangleq J_n^{(m)} \circ \Phi_n^{-1}$ for $m = 1, 2$ and $n = 1, \dots, N$; we don't get access to the true dictionary, deformations, or latent weights. Then we use the following error:

$$\mathcal{E}_{\text{cross-val}}(\alpha, \beta, \gamma) = \frac{1}{2N} \sum_{m=1}^{2} \sum_{n=1}^{N} \| W_n^{(m)} - \hat{W}_n^{(m)}[\alpha, \beta, \gamma] \|_2^2, \tag{4.12}$$

where $\hat{W}_n^{(m)}[\alpha, \beta, \gamma]$ is an estimate of warped pre-image for observation n in fold m based on the dictionary and deformations estimated using observed images from the other fold. In particular:

$$(\hat{W}_n^{(1)}[\alpha, \beta, \gamma])(x) = \begin{cases} I_n^{(1)}(x) & \text{if } x \in \text{support}_{0.75}((\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(2)}[\alpha, \beta, \gamma]) \text{ for some } k, \\ 0 & \text{otherwise,} \end{cases} \tag{4.13}$$

$$(\hat{W}_n^{(2)}[\alpha, \beta, \gamma])(x) = \begin{cases} I_n^{(2)}(x) & \text{if } x \in \text{support}_{0.75}((\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(1)}[\alpha, \beta, \gamma]) \text{ for some } k, \\ 0 & \text{otherwise,} \end{cases} \tag{4.14}$$

where $\text{support}_{0.75}(\cdot)$ is the set of voxels in an image that have absolute intensity value at least 0.75 of the maximum:

$$\text{support}_{0.75}(Z) \triangleq \{x \in \Omega : |Z(x)| \geq 0.75 \max_{y \in \Omega} |Z(y)|\}. \tag{4.15}$$

The reason why this support restriction is added is to only consider voxels that have "high enough" intensity, essentially reducing noise in the estimated warped pre-images $\hat{W}_n^{(m)}[\alpha, \beta, \gamma]$. Note that we specifically choose not to treat the dictionary and deformations learned from one fold as fixed and then estimate sparse linear combination weights for the observed images from the other fold to produce estimated warped pre-images.[1] The reason we don't use this approach is that it fails to penalize extraneous dictionary elements, which could just be assigned weight 0 for an observed image.

[1] Specifically, this approach would set estimated warped pre-image $\hat{W}_n^{(1)} \leftarrow \sum_{k=1}^{K} \hat{w}_{nk}^{(1)} (\hat{D}_k \circ \hat{\Phi}^{-1})^{(2)}[\alpha, \beta, \gamma]$, where $\hat{w}_n^{(1)} \in \mathbb{R}_+^K$ is the solution to a convex program:

$$\hat{w}_n^{(1)} \leftarrow \underset{w_n \in \mathbb{R}_+^K}{\text{argmin}} \left\{ \frac{1}{2(\hat{\sigma}^2)^{(2)}} \left\| I_n^{(1)} - \sum_{k=1}^{K} w_{nk} (\hat{D}_k \circ \hat{\Phi}^{-1})^{(2)}[\alpha, \beta, \gamma] \right\|_2^2 + \sum_{k=1}^{K} \hat{\theta}_k^{(2)} w_{nk} \right\}$$

with $(\hat{\sigma}^2)^{(2)}$ and $\hat{\theta}_k^{(2)}$ denoting estimates of parameters σ^2 and θ_k learned using data from fold 2. We can similarly define $\hat{W}_n^{(2)}$. Note that if dictionary element $\hat{D}_j^{(2)}$ is extraneous, then the convex program would set $\hat{w}_{nj}^{(1)} = 0$, which means that estimated warped pre-image $\hat{W}_n^{(1)}$ will not depend on $\hat{D}_j^{(2)}$.

We choose hyperparameters (α, β, γ) as follows:

$$(\hat{\alpha}, \hat{\beta}, \hat{\gamma}) = \underset{\alpha,\beta,\gamma \in \{0, 10^2, 10^4, 10^6\}}{\operatorname{argmin}} \mathcal{E}_{\text{cross-val}}(\alpha, \beta, \gamma). \tag{4.16}$$

Finally, we estimate dictionary $\hat{\boldsymbol{D}}$ and deformations $\hat{\boldsymbol{\Phi}}$ using all observed training data by training on images $\frac{1}{2}(I_1^{(1)} + I_1^{(2)}), \dots, \frac{1}{2}(I_N^{(1)} + I_N^{(2)})$ with hyperparameters $(\hat{\alpha}, \hat{\beta}, \hat{\gamma})$. The resulting training procedure is summarized in Alg. 8.

Algorithm 8: Training Using 2-Fold Cross-Validation with Limited Ground Truth

```
Input: Observed images $\boldsymbol{I}$
Output: Estimated dictionary $\hat{\boldsymbol{D}}$, deformations $\hat{\boldsymbol{\Phi}}$
Initialize $\mathcal{E}^{\text{opt}}_{\text{cross-val}} \leftarrow \infty, \hat{\alpha} \leftarrow 0, \hat{\beta} \leftarrow 0, \hat{\gamma} \leftarrow 0$.
for $\alpha, \beta, \gamma \in \{0, 10^2, 10^4, 10^6\}$ do
    (Fold 1) Train on images $\boldsymbol{I}^{(1)} = \{I_1^{(1)}, \dots, I_N^{(1)}\}$ using Alg. 6 with
    hyperparameters $(\alpha, \beta, \gamma)$ to produce dictionary and deformation estimates
    $(\hat{\boldsymbol{D}}^{(1)}[\alpha, \beta, \gamma], \hat{\boldsymbol{\Phi}}^{(1)}[\alpha, \beta, \gamma])$.
    (Fold 2) Train on images $\boldsymbol{I}^{(2)} = \{I_1^{(2)}, \dots, I_N^{(2)}\}$ using Alg. 6 with
    hyperparameters $(\alpha, \beta, \gamma)$ to produce dictionary and deformation estimates
    $(\hat{\boldsymbol{D}}^{(2)}[\alpha, \beta, \gamma], \hat{\boldsymbol{\Phi}}^{(2)}[\alpha, \beta, \gamma])$.
    Using $\boldsymbol{I}$, $(\hat{\boldsymbol{D}}^{(1)}[\alpha, \beta, \gamma], \hat{\boldsymbol{\Phi}}^{(1)}[\alpha, \beta, \gamma])$, and $(\hat{\boldsymbol{D}}^{(2)}[\alpha, \beta, \gamma], \hat{\boldsymbol{\Phi}}^{(2)}[\alpha, \beta, \gamma])$,
    compute $\mathcal{E}_{\text{cross-val}}(\alpha, \beta, \gamma)$, given by eq. (4.12).
    if $\mathcal{E}_{cross\text{-}val}(\alpha, \beta, \gamma) < \mathcal{E}^{opt}_{cross\text{-}val}$ then
        Set $\mathcal{E}^{\text{opt}}_{\text{cross-val}} \leftarrow \mathcal{E}_{\text{cross-val}}(\alpha, \beta, \gamma), \hat{\alpha} \leftarrow \alpha, \hat{\beta} \leftarrow \beta, \hat{\gamma} \leftarrow \gamma$.
Train on images $\{\frac{I_1^{(1)} + I_1^{(2)}}{2}, \dots, \frac{I_N^{(1)} + I_N^{(2)}}{2}\}$ using Alg. 6 with hyperparameters
$(\hat{\alpha}, \hat{\beta}, \hat{\gamma})$ to produce final dictionary and deformation estimates $(\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}})$.
```

■ 4.2.2 Heuristics in the Absence of Ground Truth

In the absence of ground truth, we use two key ideas to design heuristics for selecting hyperparameters α, β, and γ while still taking advantage of training data comprising of two separate folds. First, aligning observed images using estimated deformations should boost peak values within each estimated dictionary element support, as suggested by our toy example in Chapter 1. Second, by training separately on two different folds, the estimated dictionaries and deformations should be consistent across the two folds. We use two different consistency measures. In total, we have three heuristics, each giving a score value in $\mathbb{R}$ that quantifies the quality of a choice of hyperparameters:

- **Better alignment within dictionary elements.** We formalize the idea that aligning observed images should produce more pronounced peaks. Note that the average image $\hat{A}^{(1)}[\alpha, \beta, \gamma]$ for observed images $\boldsymbol{I}^{(1)}$ in fold 1 aligned using deformations $\hat{\boldsymbol{\Phi}}^{(2)}[\alpha, \beta, \gamma]$ learned from fold 2 is given by

$$\hat{A}^{(1)}[\alpha, \beta, \gamma](x) \leftarrow \frac{\sum_{n=1}^{N} |\mathbf{J}_{\hat{\Phi}_n^{(2)}}(x)| (I_n^{(1)} \circ \hat{\Phi}_n^{(2)}[\alpha, \beta, \gamma])(x)}{\sum_{n=1}^{N} |\mathbf{J}_{\hat{\Phi}_n^{(2)}}(x)|}. \tag{4.17}$$

To quantify the top intensity values of this average image restricted to the support of the estimated dictionary element $\hat{D}_k^{(2)}[\alpha, \beta, \gamma]$ from fold 2, we use their 75th percentile value:

$$\begin{aligned} &\hat{p}_k^{(1)}[\alpha, \beta, \gamma] \\ &\triangleq 75^{\text{th}} \text{ percentile value of } \{\hat{A}^{(1)}[\alpha, \beta, \gamma](x) : x \in \text{support}(\hat{D}_k^{(2)}[\alpha, \beta, \gamma])\}. \end{aligned} \tag{4.18}$$

We do not use the maximum, i.e., the peak value, which may be unstable.

If we did not align the images first, then the average image would be defined as

$$\bar{A}^{(1)} = \frac{1}{N} \sum_{n=1}^{N} I_n^{(1)}. \tag{4.19}$$

A conservative guess as to where estimated dictionary element $\hat{D}_k^{(2)}$ appears in $\bar{A}^{(1)}$ is to examine the union of all the supports of $\hat{D}_k^{(2)}$ deformed into the space of observation n. Thus, we quantify the top intensity values of average image $\bar{A}^{(1)}$ within the support of the k-th estimated dictionary element $\hat{D}_k[\alpha, \beta, \gamma]$ from fold 2 with:

$$\begin{aligned} &\bar{p}_k^{(1)}[\alpha, \beta, \gamma] \\ &\triangleq 75^{\text{th}} \text{ percentile value of } \{\bar{A}^{(1)}(x) : x \in \cup_{n=1}^{N} \text{support}((\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(2)}[\alpha, \beta, \gamma])\}. \end{aligned} \tag{4.20}$$

We define the improvement in top intensity value for fold 1 resulting from using information learned from fold 2 as:

$$\Delta^{(1)}[\alpha, \beta, \gamma] \triangleq \sum_{k=1}^{K} (\hat{p}_k^{(1)}[\alpha, \beta, \gamma] - \bar{p}_k^{(1)}[\alpha, \beta, \gamma]). \tag{4.21}$$

We can similarly define $\hat{A}^{(2)}, \bar{A}^{(2)}, \hat{p}_k^{(2)}, \bar{p}_k^{(2)}$, and $\Delta^{(2)}$. The average intensity improvement score across folds is thus given by

$$\mathcal{H}_{\text{intensity-improvement}}(\alpha, \beta, \gamma) = \frac{1}{2}(\Delta^{(1)}[\alpha, \beta, \gamma] + \Delta^{(2)}[\alpha, \beta, \gamma]), \tag{4.22}$$

where higher is better.

- **Consistency of alignment improvement across folds.** We seek to avoid large differences in improvement between the folds, suggesting a simple consistency score:

  $$\mathcal{H}_{\text{intensity-improvement-difference}}(\alpha, \beta, \gamma) = |\Delta^{(1)}[\alpha, \beta, \gamma] - \Delta^{(2)}[\alpha, \beta, \gamma]|, \tag{4.23}$$

  where lower is better.

- **Consistency of dictionary element supports across folds.** We want the support of dictionary elements across folds to be similar. As the dictionary element support across folds has different qualitative spaces, we compare the support across folds instead in the qualitative space of the observations. To achieve this, we measure the overlap between set $\hat{\Omega}_n^{(1)}[\alpha, \beta, \gamma] \triangleq \cup_{k=1}^K \text{support}((\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(1)}[\alpha, \beta, \gamma])$ and set $\hat{\Omega}_n^{(2)}[\alpha, \beta, \gamma] \triangleq \cup_{k=1}^K \text{support}((\hat{D}_k \circ \hat{\Phi}_n^{-1})^{(2)}[\alpha, \beta, \gamma])$ using the Dice volume overlap measure [9]:

  $$\text{Dice}(\hat{\Omega}_n^{(1)}[\alpha, \beta, \gamma], \hat{\Omega}_n^{(2)}[\alpha, \beta, \gamma]) = \frac{2|\hat{\Omega}_n^{(1)}[\alpha, \beta, \gamma] \cap \hat{\Omega}_n^{(2)}[\alpha, \beta, \gamma]|}{|\hat{\Omega}_n^{(1)}[\alpha, \beta, \gamma]| + |\hat{\Omega}_n^{(2)}[\alpha, \beta, \gamma]|}, \tag{4.24}$$

  which only measures overlap in observation n's qualitative space. Averaging across all observations' qualitative spaces gives the consistency score

  $$\mathcal{H}_{\text{dictionary-consistency}}(\alpha, \beta, \gamma) = \frac{1}{N} \sum_{n=1}^{N} \text{Dice}(\hat{\Omega}_n^{(1)}[\alpha, \beta, \gamma], \hat{\Omega}_n^{(2)}[\alpha, \beta, \gamma]), \tag{4.25}$$

  which is a value between 0 (no overlap) and 1 (perfect overlap); higher is better.

To select which hyperparameter setting to use, we handpick a hyperparameter setting that provides a good trade-off between all three heuristics.

4.3 Synthetic Data

In this section, we apply our inference algorithm to synthetic data. We describe how this data is generated in Section 4.3.1 and how we evaluate the performance of our inference algorithm in Section 4.3.2. Results are in Section 4.3.3.

4.3.1 Data Generation

Observed images, dictionary elements, and deformations are generated as follows:

Algorithm 9: Synthetic Data Generator

```
Input: Gaussian bump parameters $(\mu_1, \Sigma_1), \ldots, (\mu_{K^*}, \Sigma_{K^*})$, deformation
       generation parameters $(\sigma_v^2, \sigma_s^2)$, model parameters $(\boldsymbol{\lambda}, \sigma^2)$, model
       hyperparameter $V_{\max}$
Output: Three sets of images $\boldsymbol{I}^{(0)} = \{I_1^{(0)}, \ldots, I_N^{(0)}\}$, $\boldsymbol{I}^{(1)} = \{I_1^{(1)}, \ldots, I_N^{(1)}\}$, and
        $\boldsymbol{I}^{(2)} = \{I_1^{(2)}, \ldots, I_N^{(2)}\}$, dictionary $\boldsymbol{D} = \{D_1, \ldots, D_{K^*}$, deformations
        $\boldsymbol{\Phi} = \{\Phi_1, \ldots, \Phi_N\}$
/* Generate dictionary */
for $k = 1, \ldots, K$ do
    Set $D_k$ to be a Gaussian density with mean voxel location $\mu_k$ and covariance
    $\Sigma_k$.
    Zero out entries of $D_k$ outside the ellipsoid associated with Gaussian
    $\mathcal{N}(\mu_k, \Sigma_k)$ scaled to have maximum volume $V_{\max}$.
    Set $D_k \leftarrow D_k / \|D_k\|_2$.
/* Generate deformations */
for $n = 1, \ldots, N$ do
    Set velocity field $\mathcal{V}_n$ to consist of i.i.d. $\mathcal{N}(0, \sigma_v^2)$ entries.
    Set $\mathcal{V}_n(x)$ to be the 0 vector if voxel $x$ is not in the support of any of the
    dictionary elements.
    Apply a Gaussian-blur of variance $\sigma_s^2$ along each dimension of $\mathcal{V}_n$.
Normalize the velocity fields so that the average velocity field is 0 using
eq. (2.20) and set $\Phi_n = \exp(\mathcal{V}_n)$ for each $n$.
/* Generate observed images */
for $m = 0, 1, 2$ do
    for $n = 1, \ldots, N$ do
        Sample $w_n^{(m)} \in \mathbb{R}^{K^*}$ consisting of i.i.d. $\exp(\lambda_k)$ entries.
        Compute pre-image $J_n^{(m)} \leftarrow \sum_{k=1}^{K^*} w_{nk}^{(m)} D_k$.
        Compute observed image $I_n^{(m)} \leftarrow J_n^{(m)} \circ \Phi_n^{-1} + \varepsilon_n^{(m)}$, where $\varepsilon_n^{(m)}$ consists of
        i.i.d. $\mathcal{N}(0, \sigma^2)$ entries.
```

We specifically generate 100-by-100 images with the above procedure using parameters $\mu_1 = (45, 35)$, $\mu_2 = (40, 60)$, $\mu_3 = (65, 55)$, $\mu_4 = (60, 40)$, $\Sigma_1 = 2\mathbf{I}$, $\Sigma_2 = \mathbf{I}$, $\Sigma_3 = 3\mathbf{I}$, $\Sigma_4 = 4\mathbf{I}$, $\boldsymbol{\lambda} = (1/5, 1/8, 1/4, 1/10)$, $V^*_{\max} = 300$, $\sigma_v^2 = 4000$, $\sigma_s^2 = 36$, and $\sigma^2 = 1$. Specifying a maximum ellipse semi-axis length $r^*_{\max}$ is unnecessary as our generative process implies the existence of some $r^*_{\max}$. The dictionary generated is shown in Fig. 4.1a. Examples of the generated pre-images with their corresponding observed images are shown in Fig. 4.2. For the specific values of the parameters for σ_v^2 and σ_s^2 we use to generate these images, we find that our volume change condition is met with $\phi_{\max} = 4$.

For inference, we set the number of estimated dictionary elements to $K = 10$, the maximum ellipse volume to $V_{\max} = 500$ voxels^3, and the maximum ellipse semi-axis length to $r_{\max} = 10$ voxels. With these parameters treated as fixed, we only need to tune hyperparameters α, β, and γ.

■ 4.3.2 Evaluation

To assess error in test data, we use several error measures:

- **Deformation error.** We use error

$$\mathcal{E}_{\text{deformations}}(\hat{\boldsymbol{\Phi}}; \boldsymbol{\Phi}) = \frac{1}{N} \sum_{n=1}^{N} \sum_{x \in \Omega} \|\hat{\Phi}_n(x) - \Phi_n(x)\|_2^2. \tag{4.26}$$

- **Dictionary error.** Since the number of true dictionary elements K^* and the number of estimated dictionary elements K are, in general, not equal, we need to compute a matching between the dictionaries. For simplicity, we consider the case when $K^* < K$. Then the error we use is

$$\mathcal{E}_{\text{dictionary}}(\hat{\boldsymbol{D}}; \boldsymbol{D}) = \min_{\rho \in \mathcal{S}_K} \left\{ \sum_{k=1}^{K^*} \|\hat{D}_{\rho(k)} - D_k\|_2^2 + \sum_{k=K^*+1}^{K} \|\hat{D}_{\rho(k)}\|_2^2 \right\}, \tag{4.27}$$

  where $\mathcal{S}_K$ is the set of all permutations of $\{1, \ldots, K\}$.

  Note that for $\rho \in \mathcal{S}_K$, we can actually reorder $\rho(K^*+1), \ldots, \rho(K)$ and the objective function would be unaffected, so we're actually only optimizing over $K!/(K - K^*)!$ assignments, namely the subset of estimated dictionary elements that map to the true dictionary elements under ρ.

- **Group average error.** This error measures how far an estimate of the average group signal is from the true group average signal that perfectly accounts for misalignment. Formally, for test images $\boldsymbol{I}^{(0)} = \{I_1^{(0)}, \ldots, I_N^{(0)}\}$ with pre-images $\boldsymbol{J}^{(0)} = \{J_1^{(0)}, \ldots, J_N^{(0)}\}$, the true group average image is

$$A^{(0)} = \frac{1}{N} \sum_{n=1}^{N} J_n^{(0)}, \tag{4.28}$$

which can also be interpreted as the true group average response had we known all the deformations for aligning the images and had there been no noise.

We estimate pre-images for the test images using our dictionary and deformation estimates:

$$\hat{J}_n^{(0)}(x) = \begin{cases} (I_n^{(0)} \circ \hat{\Phi}_n)(x) & \text{if } x \text{ is in the spatial support of } \hat{D}_k \text{ for some } k, \\ 0 & \text{otherwise.} \end{cases} \tag{4.29}$$

Averaging these pre-image estimates creates

$$\hat{A}^{(0)} \leftarrow \frac{1}{N} \sum_{n=1}^{N} \hat{J}_n^{(0)}. \tag{4.30}$$

Unfortunately, images $A^{(0)}$ and $\hat{A}^{(0)}$ do not share the same qualitative space since the former has qualitative space defined by the true dictionary while the latter has one defined by the estimated dictionary. Thus, we can't compare images $A^{(0)}$ and $\hat{A}^{(0)}$ directly. However, we can bring both images $A^{(0)}$ and $\hat{A}^{(0)}$ into the qualitative space of observation n before computing the squared ℓ_2 distance between them, i.e., $\|A^{(0)} \circ \Phi_n^{-1} - \hat{A}^{(0)} \circ \hat{\Phi}_n^{-1}\|_2^2$. Repeating this distance calculation in the qualitative space of all observations $n = 1, \ldots, N$ and averaging produces the final group average error:

$$\mathcal{E}_{\text{group-average}}(\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}}; \boldsymbol{I}^{(0)}, \boldsymbol{J}^{(0)}, \boldsymbol{\Phi}) = \frac{1}{N} \sum_{n=1}^{N} \|\hat{A}^{(0)} \circ \hat{\Phi}_n^{-1} - A^{(0)} \circ \Phi_n^{-1}\|_2^2, \tag{4.31}$$

where $\hat{A}^{(0)}$ depends on $\boldsymbol{I}^{(0)}$, $\hat{\boldsymbol{D}}$, and $\hat{\boldsymbol{\Phi}}$ while $A^{(0)}$ depends on $\boldsymbol{J}^{(0)}$.

■ 4.3.3 Results

We compare our results from cross-validation training of our inference algorithm (labeled *deformation-invariant sparse coding* in tables and plots) against the baseline of the exact same approach except where all the estimated deformations are constrained to be identity (labeled *sparse coding* in tables and plots). We also compare against another baseline that does not use deformations; specifically we simply estimate the dictionary elements by using the dictionary initialization procedure described in Section 4.1 except that rather than performing serial groupwise registration in the first step to obtain an average image, the average image is estimated to be $\frac{1}{2N}\sum_{m=1}^{2}\sum_{n=1}^{N} I_n^{(m)}$. (We label this second baseline method as *watershed* in tables and plots.)

Estimated dictionaries by the three methods are shown in Figs. 4.1b, 4.1c, and 4.1d. Strictly for display purposes, the estimated dictionary elements within each method are permuted so that dictionary elements across methods correspond visually, and we normalize the intensity of each dictionary element. While the watershed baseline is intentionally initialized to have the same number of dictionary elements as the ground truth, the sparse coding baseline and deformation-invariant sparse coding both estimate four non-zero dictionary elements, also agreeing with the ground truth, even though they were initialized with 10 dictionary elements each. Visually, deformation-invariant sparse coding finds dictionary elements that are the most "concentrated" across the three methods. In other words, the region containing the main peak within each dictionary element is less spread out.

Deformation, dictionary, and group average errors are reported in Table 4.1. We remark that we provide no guarantee that our inference algorthm produces unbiased estimates for the dictionary or the deformations. Specifically, the qualitative space of the estimated dictionary is not guaranteed to match the qualitative space of the true dictionary, which explains why we do not expect the deformation error or the dictionary error to be lower for deformation-invariant sparse coding compared to those of the baseline methods. However, as the group average error across the three methods suggests, while the qualitative space of the true dictionary is not recovered, the estimated group average signal for deformation-invariant sparse coding is substantially closer to the true group average signal compared to the group average signal achieved by not accounting for deformations. These average signals of test images with and without pre-aligning with estimated deformations are shown in Fig. 4.3, where using deformations results in more pronounced peaks. This is promising since for real data, estimating the group average signal is ultimately what we want rather than exactly recovering the qualitative space of the group average signal.

Next, we verify that trading off the heuristics in Section 4.2.2 for selecting hyperparameters in the absence of ground truth can yield a good hyperparameter selec-

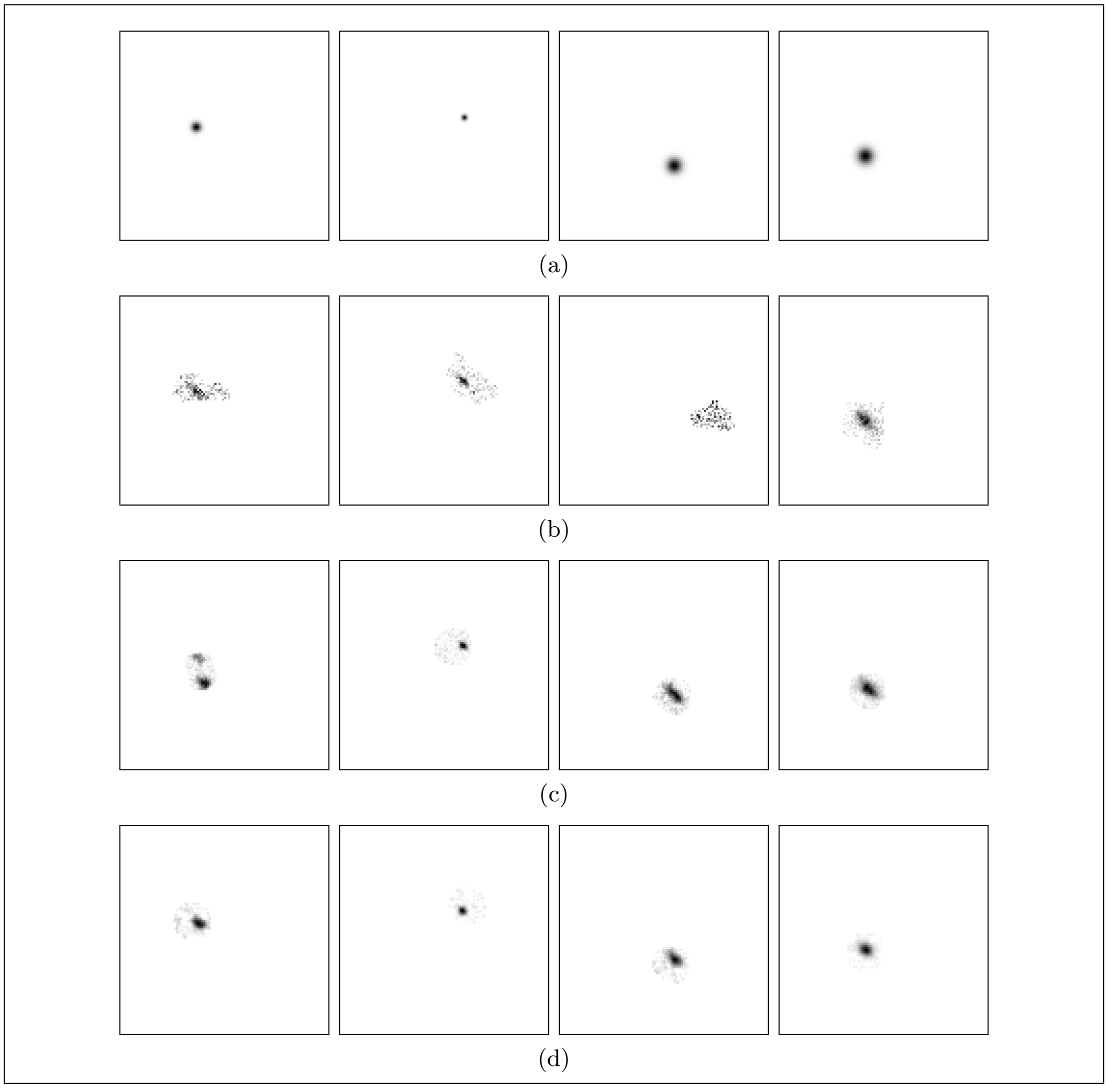


Figure 4.1: The synthetic data's (a) true dictionary, and estimated dictionaries using (b) the watershed baseline, (c) the sparse coding baseline, and (d) deformation-invariant sparse coding.

Method	$\mathcal{E}_{\text{deformations}}$	$\mathcal{E}_{\text{dictionary}}$	$\mathcal{E}_{\text{group-average}}$
Watershed	17811.8297	4.2473	166.9231
Sparse coding	17811.8297	3.1564	169.2603
Deformation-invariant sparse coding	45737.5875	2.8732	150.2779

Table 4.1: Deformation, dictionary, and group average errors across different methods.

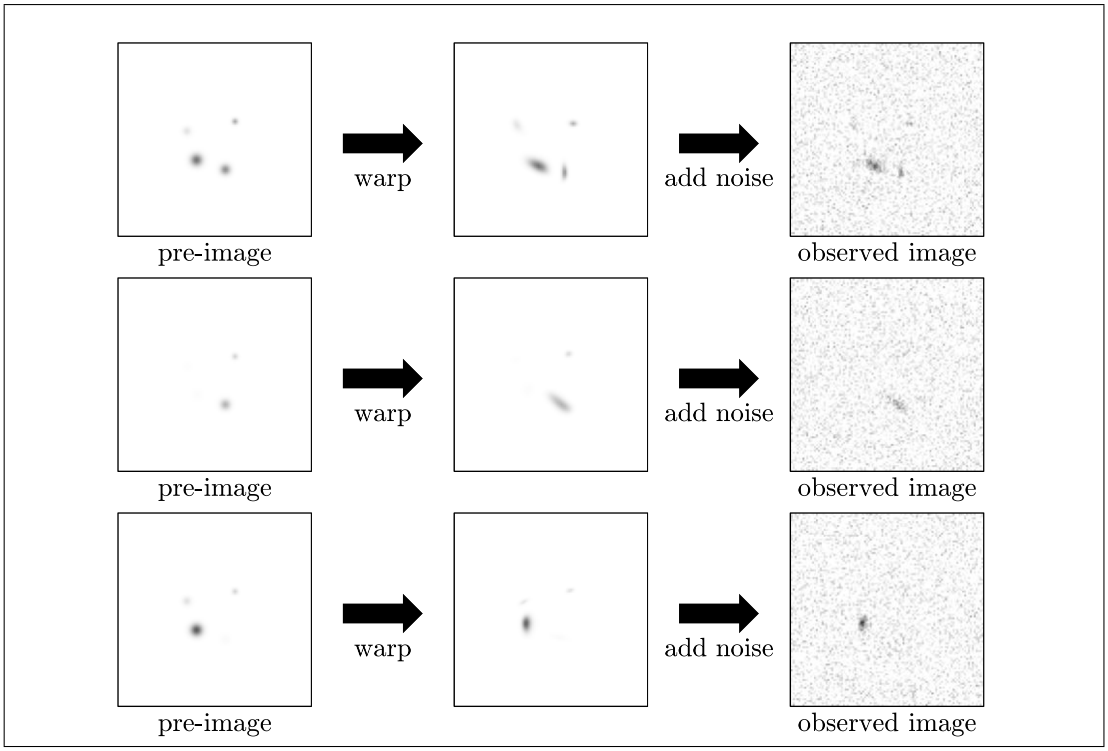


Figure 4.2: Synthetic data examples of pre-images and their corresponding observed images. All images are shown with the same intensity scale.

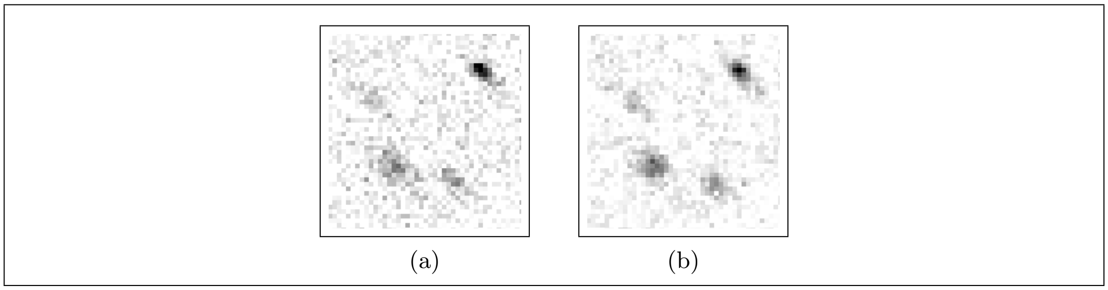


Figure 4.3: Synthetic data average of test images (a) without deformations, and (b) aligned with deformations estimated from training data using deformation-invariant sparse coding. Both images are shown with the same intensity scale and zoomed in.

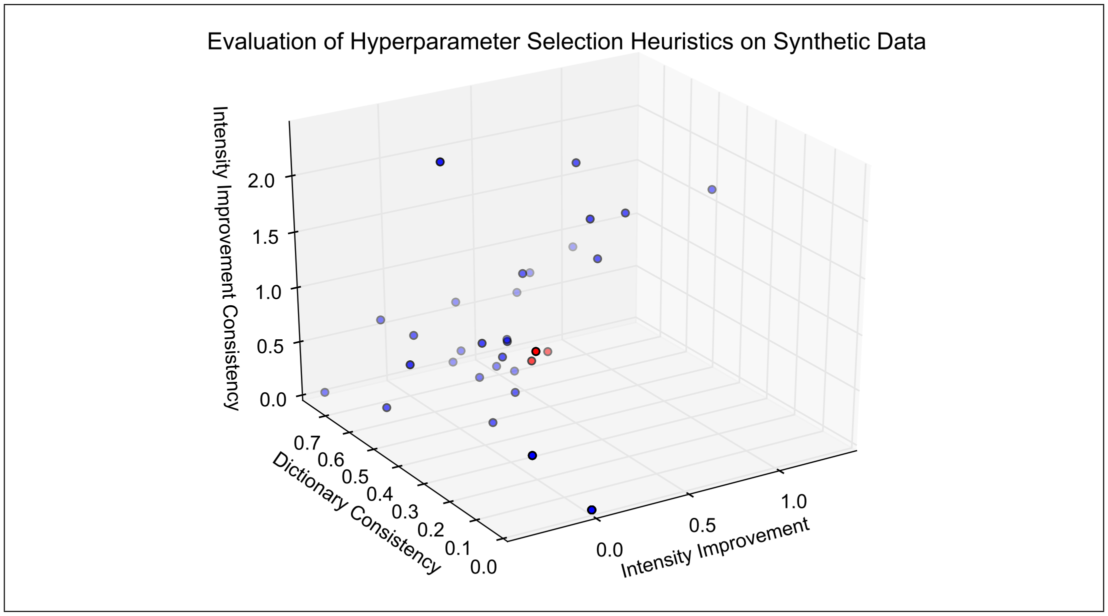


Figure 4.4: Scatter plot used to evaluate hyperparameter selection heuristics for synthetic data. Each hyperparameter setting (α, β, γ) is depicted as a single point, where $\alpha, \beta, \gamma \in \{0, 10^2, 10^4, 10^6\}$. The top 5 hyperparameter settings (some overlap in the plot) according to our group average error on test data are shown in red; all other hyperparameter settings are shown in blue.

tion. We show a scatter plot across the three different heuristics' scores in Fig. 4.4, where each point corresponds to a hyperparameter configuration (α, β, γ) for $\alpha, \beta, \gamma \in \{0, 10^2, 10^4, 10^6\}$; red points indicate the top 5 hyperparameter configurations according to group average error on test data, and blue points are all other hyperparameter configurations. As seen in the scatter plot, the best hyperparameter configurations according to our group average error correspond to configurations that simultaneously strike a balance between high dictionary consistency, low intensity improvement difference, and intensity improvement that is not too low.

■ 4.4 Language fMRI Study

We apply our inference algorithm on real fMRI data from a language fMRI study. Importantly, this data does not have any ground truth. We provide some details on the dataset in Section 4.4.1, discuss how we evaluate our algorithm in Section 4.4.2, and present results in Section 4.4.3.

4.4.1 Data

Our dataset is from an fMRI study of 82 subjects reading sentences and pronounceable non-words [14]. First, we apply the standard fMRI general linear model [16] and weighted random effects analysis [33] for the sentences vs. non-words contrast, which for the purposes of this thesis amounts to applying a black box that takes as input raw fMRI time course data from a subject and outputs a t-statistic map. In one of these maps, a voxel, which corresponds to a location in the brain, has a t-statistic indicating statistical significance to the sentences vs. non-words contrast, which serves as an indicator for lexical and structural processing. Observed images $\boldsymbol{I}$ are thus taken to be the t-statistic maps of each subject thresholded at p-value=0.01, where each subject's t-statistic map has been affinely pre-aligned to the MNI305 template brain [13] based on the corresponding anatomical MRI scan as to account for anatomical variability.

As with the synthetic data setting, subject n has three observed images $I_n^{(0)}, I_n^{(1)}, I_n^{(2)}$, each originating from a separate run of the fMRI protocol where the subject was asked to essentially repeat the same language task. Each image $I_n^{(m)}$ is of size $128 \times 128 \times 128$, where each voxel is of $(2\text{mm})^3$ volume. We train on images $\boldsymbol{I}^{(1)} = \{I_1^{(1)}, \ldots, I_N^{(1)}\}$ and $\boldsymbol{I}^{(2)} = \{I_1^{(2)}, \ldots, I_N^{(2)}\}$ and test on images $\boldsymbol{I}^{(0)} = \{I_1^{(0)}, \ldots, I_N^{(0)}\}$. For inference, we set $K = 20$, $V_{\max} = 1000$, and $r_{\max} = 7$. Empirically, we find that the volume change condition is satisfied with $\phi_{\max} = 2$.

4.4.2 Evaluation

Due to the lack of ground truth, we can only examine the learned dictionary elements and qualitatively compare against existing neuroscience literature. As for validating the deformations, we work off the intuition that accounting for deformations should make the peaks within group-level parcels more pronounced. To this end, we pre-align raw fMRI data associated with test images $\boldsymbol{I}^{(0)}$ using deformations learned from training images $(\boldsymbol{I}^{(1)}, \boldsymbol{I}^{(2)})$ and actually re-run standard fMRI analysis to produce new t-statistic maps, which now account for the deformations learned; this essentially amounts to applying deformation Φ_n to $I_n^{(0)}$ for each subject n. Then we want to look within each dictionary element support to see if there are more pronounced peaks.

Specifically for drawing conclusions on the group-level parcels defined by the estimated dictionary elements, within each parcel, it is the peak and regions around the peak that are of interest rather than the full support of the dictionary element. Thus, to quantify the advantage of pre-aligning with our estimated deformations, within each dictionary element we compare the top 25% highest significance values for our method versus those of anatomical alignment.[2] Formally, we compare average image

$$\bar{A}^{(0)} \leftarrow \frac{1}{N} \sum_{n=1}^{N} I_n^{(0)}, \tag{4.32}$$

which has only accounted for anatomical alignment, with average image

$$\hat{A}^{(0)} \leftarrow \frac{1}{N} \sum_{n=1}^{N} (I_n^{(0)} \circ \hat{\Phi}_n), \tag{4.33}$$

where, as discussed above, $I_n^{(0)} \circ \hat{\Phi}_n$ is actually computed by pre-aligning raw fMRI data and re-running standard fMRI analysis rather than directly deforming $I_n^{(0)}$. Then to compare the top 25% of significance values in the support of estimated dictionary element $\hat{D}_k$, we plot a histogram of the top 25% highest values in $\{\hat{A}(x) : x \in \text{support}(\hat{D}_k)\}$ versus the top 25% highest values in $\{\bar{A}(x) : x \in \cup_{n=1}^{N} \text{support}(\hat{D}_k \circ \hat{\Phi}_n^{-1})\}$. For visualization purposes, we shall show the histograms as box plots, where rather than using the t-statistic values, we actually show the negative log p-values associated with each of the t-statistic values. Negative log p-values have a natural interpretation of statistical significance, with an increase of 1 negative log p-value meaning that the p-value dropped by an order of magnitude. Of course, we can then look at how much different dictionary elements benefit from the estimated deformations.

■ 4.4.3 Results

Using our approach from Section 4.2.2 to select hyperparameters is expensive as it requires iterating through many combinations of hyperparameters. As such, we only computed our heuristics on an arbitrary choice of 20 subjects to obtain the scatter plot in Fig. 4.5, where each point corresponds to a specific choice of hyperparameters $(\alpha, \beta, \gamma) \in \{0, 10^2, 10^4, 10^6, 10^8\}^3$. Points that we deemed to be a good-trade off between the three different heuristics are shown in green and correspond to hyperparameter settings $(\alpha, \beta, \gamma) = (10^4, 0, 0), (10^4, 0, 10^2), (10^4, 10^2, 0), (10^4, 10^2, 10^2), (10^4, 10^4, 0), (10^4, 10^4, 10^2), (10^4, 10^4, 10^4)$. With these hyperparameter settings, we then estimated dictionaries and deformations using all 82 subjects to find that the results were similar. Thus, in what

[2] We've found that looking at the top 50% up through the top 1% of the highest significance values yields similar results, so the choice of looking specifically at the top 25% is arbitrary but sufficient for our purposes.

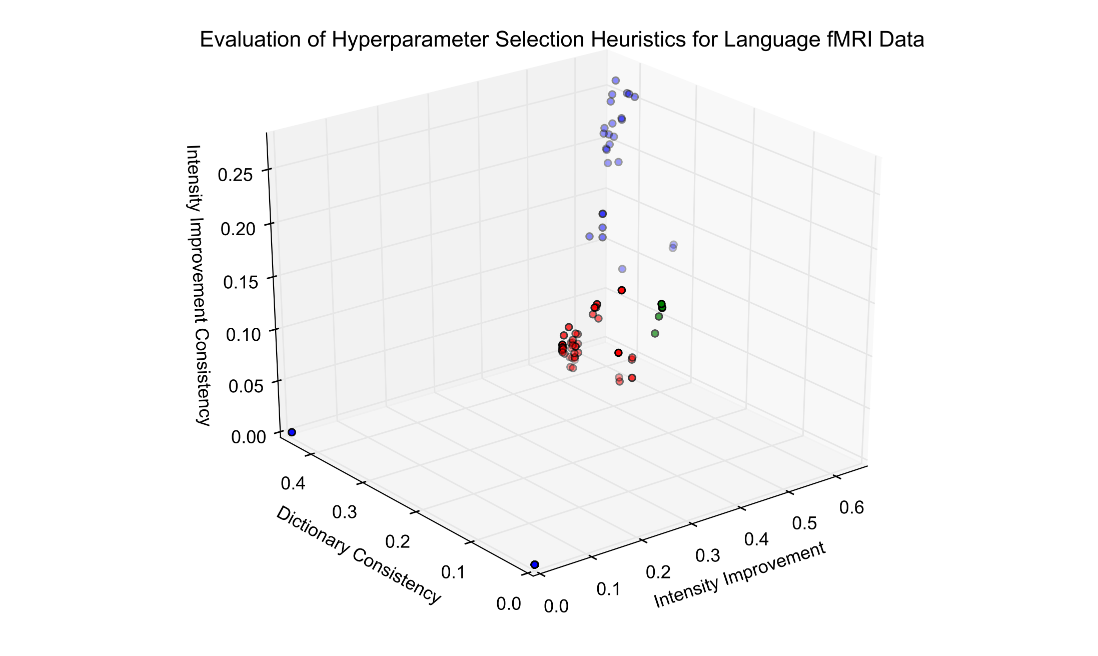
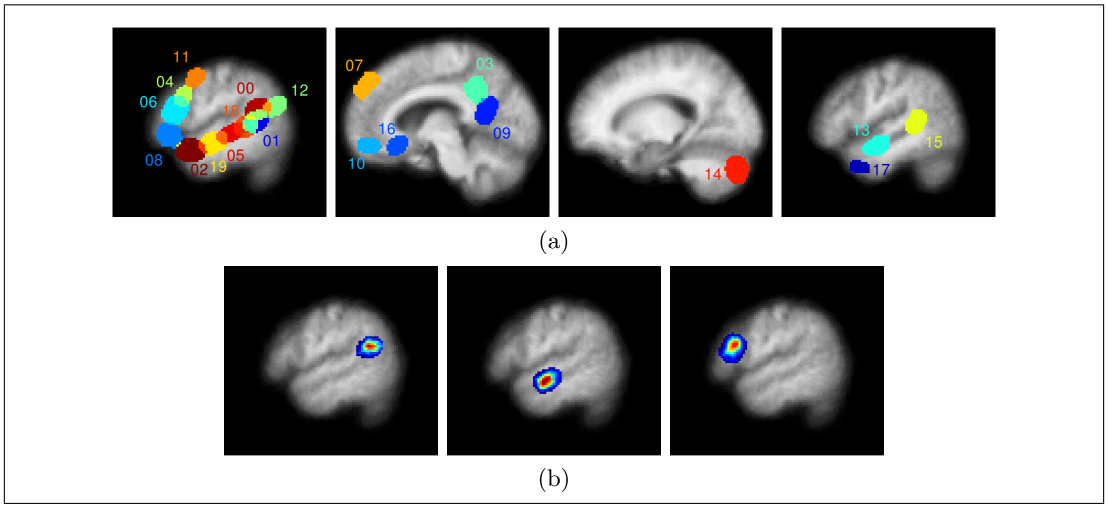


Figure 4.5: Scatter plot used to evaluate hyperparameter selection heuristics for language fMRI data. Each hyperparameter setting (α, β, γ) is depicted as a single point, where $\alpha, \beta, \gamma \in \{0, 10^2, 10^4, 10^6, 10^8\}$. Hyperparameter settings in red and green have comparable dictionary consistency, $\mathcal{H}_{\text{intensity-improvement-difference}} < 0.1$, and $\mathcal{H}_{\text{intensity-improvement}} > 0.4$. Hyperparameter settings in green further achieve $\mathcal{H}_{\text{intensity-improvement}} > 0.6$.

follows, we only show results for hyperparameter setting $\alpha = \beta = \gamma = 10^4$.

Fig. 4.6a shows the spatial support of the final learned dictionary elements on four slices. Fig. 4.6b illustrates some of the dictionary elements extracted by the algorithm. The dictionary elements include regions previously reported as indicative of lexical and structural language processing [14], namely portions of the temporal lobes, the right cerebellum, and the left frontal lobe. There are also dictionary elements corresponding to the medial prefrontal cortex, the posterior cingulate, and the precuneus.

Next, we validate the estimated deformations using the method described in Section 4.4.2. Within each estimated dictionary element/group-level parcel, we compare the top 25% highest significance values for our method versus those of anatomical alignment; the resulting box plots are shown in Fig. 4.7. We observe that accounting for functional variability via deformations results in higher peak significance values within the estimated group-level parcels, suggesting better overlap of these functional activa-



(a)

(b)

Figure 4.6: Estimated dictionary. (a) Four slices of a map showing the spatial support of the extracted dictionary elements. Different colors correspond to distinct dictionary elements where there is some overlap between dictionary elements. From left to right: left frontal lobe and temporal regions, medial prefrontal cortex and posterior cingulate/precuneus, right cerebellum, and right temporal lobe. Dictionary element indices correspond to those in Fig. 4.7. (b) A single slice from three different estimated dictionary elements where relative intensity varies from high (red) to low (blue). From left to right: left posterior temporal lobe, left anterior temporal lobe, left inferior frontal gyrus.

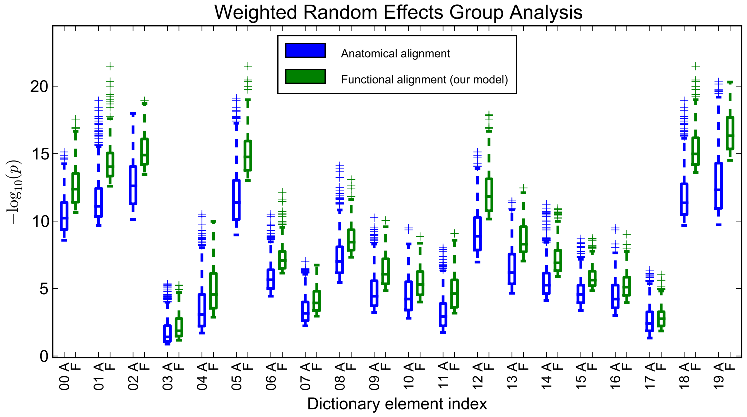


Figure 4.7: Box plots of top 25% weighted random effects analysis significance values within dictionary element supports. For each dictionary element, “A” refers to anatomical alignment, and “F” refers to alignment via deformations learned by our model.

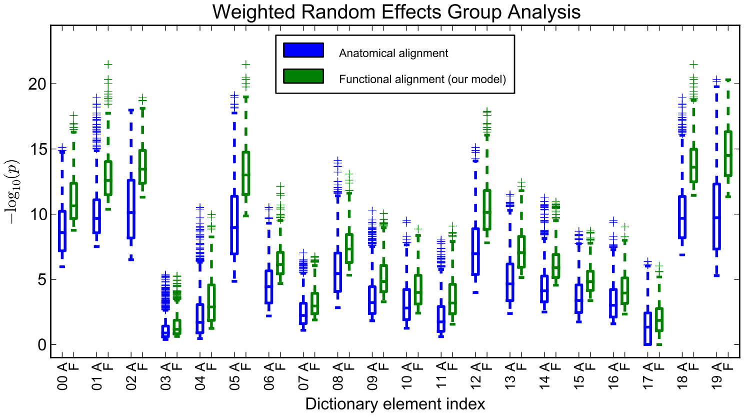


Figure 4.8: Box plots of top 50% weighted random effects analysis significance values within dictionary element supports. For each dictionary element, "A" refers to anatomical alignment, and "F" refers to alignment via deformations learned by our model.

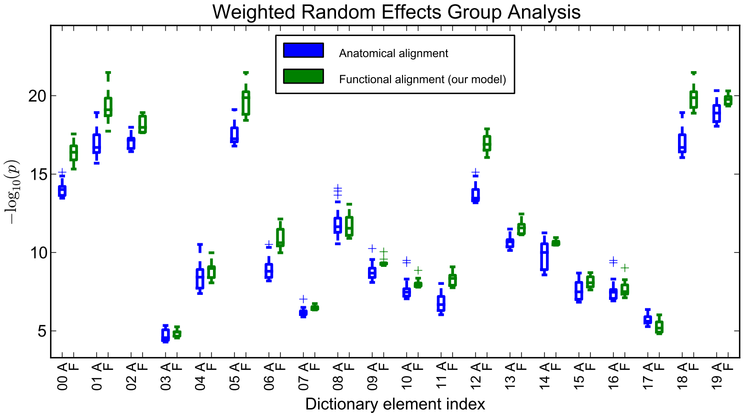


Figure 4.9: Box plots of top 1% weighted random effects analysis significance values within dictionary element supports. For each dictionary element, "A" refers to anatomical alignment, and "F" refers to alignment via deformations learned by our model.

Dict. elt. index	Improvement	Description
19	4.78	Left mid anterior temporal lobe
5	4.03	Left mid temporal lobe
18	3.92	Left mid posterior temporal lobe
2	3.34	Left anterior temporal lobe
12	3.19	Left posterior temporal lobe
1	2.91	Left posterior temporal lobe
13	2.39	Right mid anterior temporal lobe
0	2.06	Left posterior temporal lobe
8	1.88	Left inferior frontal gyrus orbital
14	1.77	Right cerebellum
6	1.71	Left inferior frontal gyrus
9	1.62	Posterior cingulate/precuneus
15	1.46	Right mid posterior temporal lobe
11	1.44	Left middle frontal gyrus
10	1.20	Ventral medial prefrontal cortex
4	1.18	Left inferior frontal gyrus
16	0.94	Ventral medial prefrontal cortex
7	0.72	Dorsal medial prefrontal cortex
17	0.52	Right anterior temporal lobe
3	0.29	Posterior cingulate/precuneus

Table 4.2: Group-level parcels ranked by improvement in 75th percentile $-\log(p\text{-value})$.

tion regions across subjects. On average, our method improves the significance of group analysis by roughly 1.5 orders of magnitude when looking at the top 25% significance values. Similar results hold when looking at the top 50% or the top 1% of significance values as shown in Figs. 4.8 and 4.9, respectively.

Lastly, by cross-referencing with Fig. 4.6a to associate dictionary element/group-parcel indices with activated brain regions, we can identify which functional regions benefit the most or the least from the estimated deformations. The full ranking is shown in Table 4.2. The left temporal lobe benefits the most from the estimated deformations, suggesting that it has significant functional variability. Meanwhile, the medial prefrontal cortex, right anterior temporal lobe, and part of the posterior cingulate/precuneus benefit the least from estimated deformations.

Chapter 5

Discussion and Conclusions

This thesis has extended sparse coding to account for deformations and provided an accompanying inference algorithm that can take advantage of existing work in image registration. Our treatment has largely been algorithmic rather than theoretical. Natural questions that arise are how inference can be changed to provide theoretical guarantees for consistently estimating dictionary and deformations, and how much observed data is needed to obtain accurate dictionary and deformation estimates. The latter question has practical implications as to determine how many subjects are needed for an fMRI study if deformation-invariant sparse coding is to be used. The former question remains open; Kurtek *et al.* [20] have recently resolved the case of a single dictionary element and noise that is a single unknown constant across observations, but analyzing our more general setting has yet to be done.

On the neuroscience side, more validation is needed to justify the usefulness of modeling functional variability using deformation-invariant sparse coding. A direction worth exploring is to see whether an estimated dictionary learned from, say, a language study can be used as markers for a new study with a different stimulus to see how the new stimulus relates to language processing. If the estimated dictionary indeed provides a more accurate assessment of the stimulus used to train the dictionary than other existing approaches, then we could confirm the utility of our model and hopefully use it for neuroscientific discovery.

Appendix A

Deriving the Inference Algorithm

We derive our EM-like inference algorithm in Chapter 3. Specifically, we would like to solve optimization problem (3.2), which is equivalent to solving

$$(\hat{\boldsymbol{D}}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{\lambda}}, \hat{\sigma}^2) \leftarrow \underset{\boldsymbol{D},\boldsymbol{\Phi},\boldsymbol{\lambda},\sigma^2}{\operatorname{argmin}} \int p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) d\boldsymbol{w}. \tag{A.1}$$

The full joint distribution in the integrand is given by eq. (3.6), reproduced below for convenience:

$$\begin{aligned} & p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) \\ & \propto \prod_{n=1}^{N} \left\{ p_\Phi(\Phi_n) \prod_{k=1}^{K} p_w(w_{nk}; \lambda_k) \prod_{x \in \Omega} \mathcal{N}\left(I_n(x); \sum_{k=1}^{K} w_{nk} D_k(\Phi_n^{-1}(x)), \sigma^2 \right) \right\} \\ & \quad \cdot \exp \left\{ -\sum_{k=1}^{K} (\alpha \|D_k\|_1 + \frac{\beta}{2} D_k^\top \mathbf{L} D_k) - \gamma \sum_{k \neq \ell} \|D_k \odot D_\ell\|_1 \right\}, \end{aligned} \tag{A.2}$$

where average deformation $\Phi_1 \circ \cdots \circ \Phi_n$ is identity; each D_k satisfies $\|D_k\|_2 \leq 1$ and has spatial support contained within an ellipsoid of volume $V_{\max}$ and maximum semi-axis length $r_{\max}$; and hyperparameters α, β, γ, $V_{\max}$, and $r_{\max}$ are treated as constants. Distributions $p_w(\cdot; \lambda_k)$ and $p_\Phi(\cdot)$ are left general. We specialize to the case of exponential p_w and positive normal q_w in Appendix B, which builds off the results of this section. Importantly, our derivations in this section work with any registration framework that minimizes an energy of the form in eq. (3.3).

Since marginalizing out the latent weights $\boldsymbol{w}$ in optimization problem (A.1) is intractable, the EM algorithm instead locally maximizes a lower bound on log likelihood $\log p(\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)$. In particular, using Jensen's inequality and introducing an auxiliary distribution $q(\boldsymbol{w})$ over latent weights $\boldsymbol{w}$, we have

$$\begin{aligned}
\log p(\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) &= \log \int q(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2) \frac{p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)}{q(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)} d\boldsymbol{w} \\
&= \log \mathbb{E}_q \left[\frac{p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)}{q(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)} \middle| \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D} \right] \\
&\geq \mathbb{E}_q \left[\log \frac{p(\boldsymbol{I}, \boldsymbol{w}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)}{q(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)} \middle| \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D} \right] \\
&= \sum_{n=1}^{N} \Bigg[\log p_\Phi(\Phi_n) + \sum_{k=1}^{K} \mathbb{E}_q[\log p_w(w_{nk}; \lambda_k) | \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}] \\
&\qquad - \frac{1}{2\sigma^2} \mathbb{E}_q \left[\left\| I_n - \sum_{k=1}^{K} w_{nk} (D_k \circ \Phi_n^{-1}) \right\|_2^2 \middle| \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D} \right] \Bigg] \\
&\quad - \frac{N|\Omega|}{2} \log(2\pi\sigma^2) + H(q) + \log p(\boldsymbol{D}) + \text{constant} \\
&\triangleq \mathcal{L}(q, \boldsymbol{\Phi}, \boldsymbol{D}, \boldsymbol{\lambda}, \sigma^2), \qquad \text{(A.3)}
\end{aligned}$$

where $H(q)$ is the differential entropy of distribution $q(\boldsymbol{w})$. The EM algorithm iteratively maximizes this lower bound $\mathcal{L}(q, \boldsymbol{\Phi}, \boldsymbol{D}, \boldsymbol{\lambda}, \sigma^2)$ via coordinate ascent until reaching a local maximum. Exact EM calls for choosing $q(\boldsymbol{w}) = p(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)$; however, with this choice of q and by inspecting the graphical model (Fig. 3.2), once we condition on observed images $\boldsymbol{I}$, all the weights w_{nk} for the same n become coupled, rendering expectation $\mathbb{E}_q[\|I_n - \sum_{k=1}^{K} w_{nk}(D_k \circ \Phi_n^{-1})\|_2^2 | \boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}]$ intractable to compute due to the marginalizations required. Thus, we choose auxiliary distribution q to be a variational approximation to the distribution $p(\boldsymbol{w}|\boldsymbol{I}, \boldsymbol{\Phi}, \boldsymbol{D}; \boldsymbol{\lambda}, \sigma^2)$ where q has factorization

$$q(\boldsymbol{w}; \boldsymbol{\psi}) = \prod_{n=1}^{N} \prod_{k=1}^{K} q_w(w_{nk}; \psi_{nk}), \qquad \text{(A.4)}$$

and distribution $q_w(\cdot; \psi_{nk})$ is parameterized by ψ_{nk}. With this choice of q and a bit of algebra, lower bound $\mathcal{L}$ in inequality (A.3) can be written as

$$\begin{aligned}
&\mathcal{L}(q(\cdot;\boldsymbol{\psi}),\boldsymbol{\Phi},\boldsymbol{D},\boldsymbol{\lambda},\sigma^2)\\
&=\sum_{n=1}^{N}\Bigg[\log p_{\Phi}(\Phi_n)+\sum_{k=1}^{K}\mathbb{E}_{q_w}[\log p_w(w_{nk};\lambda_k)|\boldsymbol{I},\boldsymbol{\Phi},\boldsymbol{D}]\\
&\qquad\qquad-\frac{1}{2\sigma^2}\Bigg[\Bigg\|I_n-\sum_{k=1}^{K}\langle w_{nk}\rangle(D_k\circ\Phi_n^{-1})\Bigg\|_2^2+\sum_{k=1}^{K}(\langle w_{nk}^2\rangle-\langle w_{nk}\rangle^2)\|D_k\circ\Phi_n^{-1}\|_2^2\Bigg]\Bigg]\\
&\quad-\frac{N|\Omega|}{2}\log(2\pi\sigma^2)+\sum_{n=1}^{N}\sum_{k=1}^{K}H(q_w(\cdot;\psi_{nk}))+\log p(\boldsymbol{D})+\text{constant}\\
&\triangleq\mathcal{L}^{-}(\boldsymbol{\psi},\boldsymbol{\Phi},\boldsymbol{D},\boldsymbol{\lambda},\sigma^2),
\end{aligned}\tag{A.5}$$

where $\langle w_{nk}\rangle\triangleq\mathbb{E}_q[w_{nk}|\boldsymbol{I},\boldsymbol{\Phi},\boldsymbol{D}]$ and $\langle w_{nk}^2\rangle\triangleq\mathbb{E}_q[w_{nk}^2|\boldsymbol{I},\boldsymbol{\Phi},\boldsymbol{D}]$.

Since q is chosen to not necessarily be the optimal choice $q(\boldsymbol{w})=p(\boldsymbol{w}|\boldsymbol{I},\boldsymbol{\Phi},\boldsymbol{D};\boldsymbol{\lambda},\sigma^2)$, the E-step of the resulting EM-like algorithm does not guarantee maximization of lower bound $\mathcal{L}$ of ineq. (A.3) over all possible distributions for latent weights $\boldsymbol{w}$ but instead maximizes the looser variational lower bound $\mathcal{L}^{-}$. This approximation results in a variational EM algorithm [38]. For part of the M-step, we maximize an even looser approximate lower bound on the log likelihood. Effectively, our inference algorithm is based on the EM algorithm but lacks theoretical guarantees of the latter due to approximations we make. Moreover, maximizing looser bounds does not guarantee an increase in the log likelihood at each step, so our algorithm is not a generalized EM algorithm.

Equipped with variational lower bound $\mathcal{L}^{-}$, we're ready to derive the steps of each iteration of our inference algorithm. First, we let $(\hat{\boldsymbol{\Phi}},\hat{\boldsymbol{D}},\hat{\boldsymbol{\lambda}},\hat{\sigma}^2,\hat{\boldsymbol{\psi}})$ be current estimates for $(\boldsymbol{\Phi},\boldsymbol{D},\boldsymbol{\lambda},\sigma^2,\boldsymbol{\psi})$ and denote $\langle\hat{w}_{nk}\rangle\triangleq\mathbb{E}_{\hat{q}_w}[w_{nk}|\boldsymbol{I},\hat{\boldsymbol{\Phi}},\hat{\boldsymbol{D}}]$, $\langle\hat{w}_{nk}^2\rangle\triangleq\mathbb{E}_{\hat{q}_w}[w_{nk}^2|\boldsymbol{I},\hat{\boldsymbol{\Phi}},\hat{\boldsymbol{D}}]$, and $\hat{q}_w=q_w(\cdot;\hat{\psi}_{nk})$. Then our EM-like algorithm alternates between the following steps:

E-step. Compute

$$\hat{\boldsymbol{\psi}}\leftarrow\underset{\boldsymbol{\psi}}{\operatorname{argmax}}\ \mathcal{L}^{-}(\boldsymbol{\psi},\hat{\boldsymbol{\Phi}},\hat{\boldsymbol{D}},\hat{\boldsymbol{\lambda}},\hat{\sigma}^2).\tag{A.6}$$

Then compute expectations $\langle\hat{w}_{nk}\rangle$ and $\langle\hat{w}_{nk}^2\rangle$ for all n and k.

M-step. Compute

$$(\hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}, \hat{\boldsymbol{\lambda}}, \hat{\sigma}^2) \leftarrow \underset{\boldsymbol{\Phi},\boldsymbol{D},\boldsymbol{\lambda},\sigma^2}{\operatorname{argmax}} \mathcal{L}^-(\hat{\boldsymbol{\psi}}, \boldsymbol{\Phi}, \boldsymbol{D}, \boldsymbol{\lambda}, \sigma^2), \tag{A.7}$$

which depends on expectations $\langle \hat{w}_{nk} \rangle$ and $\langle \hat{w}_{nk}^2 \rangle$.

The next sections detail how each step is carried out.

A.1 E-step

Rather than simultaneously updating all estimates $\hat{\psi}_{nk}$ to numerically optimize (A.6), we do a coordinate ascent, i.e., we maximize variational lower bound $\mathcal{L}^-$ with respect to ψ_{nk}. To do this, we'll make use of the following identity:

$$\begin{aligned}
&\left\| I_n - \sum_{k=1}^{K} \langle w_{nk} \rangle (D_k \circ \Phi_n^{-1}) \right\|_2^2 + (\langle w_{nk}^2 \rangle - \langle w_{nk} \rangle^2) \| D_k \circ \Phi_n^{-1} \|_2^2 \\
&= \sum_{x \in \Omega} \Bigg[\left(\left(I_n(x) - \sum_{\ell \neq k} \langle w_{n\ell} \rangle D_\ell(\Phi_n^{-1}(x)) \right) - \langle w_{nk} \rangle D_k(\Phi_n^{-1}(x)) \right)^2 \\
&\qquad + (\langle w_{nk}^2 \rangle - \langle w_{nk} \rangle^2) D_k^2(\Phi_n^{-1}(x)) \Bigg] \\
&= \sum_{x \in \Omega} \Bigg[\left(I_n(x) - \sum_{\ell \neq k} \langle w_{n\ell} \rangle D_\ell(\Phi_n^{-1}(x)) \right)^2 \\
&\qquad - 2 \left(I_n(x) - \sum_{\ell \neq k} \langle w_{n\ell} \rangle D_\ell(\Phi_n^{-1}(x)) \right) \langle w_{nk} \rangle D_k(\Phi_n^{-1}(x)) + \langle w_{nk}^2 \rangle D_k^2(\Phi_n^{-1}(x)) \Bigg].
\end{aligned} \tag{A.8}$$

Dropping terms in $\mathcal{L}^-$—viz. eq. (A.5)—that do not involve ψ_{nk} for fixed n and k, and using eq. (A.8), we see that we should set estimate $\hat{\psi}_{nk}$ to be the minimizer of energy

$$\begin{aligned}
&E(\psi_{nk}) \\
&= -\mathbb{E}_{q_w}[\log p_w(w_{nk}; \hat{\lambda}_k) | \boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}] \\
&\quad - \frac{1}{\hat{\sigma}^2} \sum_{x \in \Omega} \left[\left(I_n(x) - \sum_{\ell \neq k} \langle \hat{w}_{n\ell} \rangle \hat{D}_\ell(\hat{\Phi}_n^{-1}(x)) \right) \langle w_{nk} \rangle D_k(\Phi_n^{-1}(x)) - \frac{1}{2} \langle w_{nk}^2 \rangle D_k^2(\Phi_n^{-1}(x)) \right] \\
&\quad - H(q_w(\cdot; \psi_{nk})), \qquad \text{(A.9)}
\end{aligned}$$

where $\langle w_{nk} \rangle$ and $\langle w_{nk}^2 \rangle$ depend on ψ_{nk} as they are expectations with respect to $q_w(\cdot; \psi_{nk})$.

Another interpretation makes it clear that the E-step involves variational inference, which will be handy for derivations in Appendix B. In particular, note that all terms except the trailing entropy term in energy (A.9) are from the expected log posterior of w_{nk} given $I_n, \hat{\Phi}_n, \hat{\boldsymbol{D}}$, and $\langle \hat{w}_{n\ell} \rangle$ for $\ell \neq k$; the expectation is taken over random variable w_{nk} with respect to $q_w(w_{nk}; \psi_{nk})$; we denote this posterior as $p_w(w_{nk} | I_n, \langle \hat{w}_{n\neg k} \rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$. Indeed, a few lines of algebra shows that if we take joint distribution (3.6), fix everything except w_{nk}, and use eq. (A.8), then we obtain proportionality

$$\begin{aligned}
&p_w(w_{nk} | I_n, w_{n\neg k}, \Phi_n, \boldsymbol{D}; \lambda_k, \sigma^2) \\
&\propto p_w(w_{nk}; \lambda_k) \times \\
&\quad \exp\left\{ \frac{1}{\sigma^2} \sum_{x \in \Omega} \left[\left(I_n(x) - \sum_{\ell \neq k} w_{n\ell} D_\ell(\Phi_n^{-1}(x)) \right) w_{nk} D_k(\Phi_n^{-1}(x)) - \frac{1}{2} w_{nk}^2 D_k^2(\Phi_n^{-1}(x)) \right] \right\}. \\
&\qquad \text{(A.10)}
\end{aligned}$$

Plugging in expectations $\langle \hat{w}_{n\ell} \rangle$ in place of $w_{n\ell}$ for $\ell \neq k$ and estimates $(\hat{\Phi}_n, \hat{\boldsymbol{D}}, \hat{\lambda}_k, \hat{\sigma}^2)$ in place of $(\Phi_n, \boldsymbol{D}, \lambda_k, \sigma^2)$, we can then take logs, apply expectation over random variable w_{nk} with respect to $q_w(w_{nk}; \psi_{nk})$, and negate both sides to precisely recover the first two terms of (A.9). We thus arrive at the critical observation that

$$\begin{aligned}
E(\psi_{nk}) &= -\mathbb{E}_{q_w}[\log p_w(w_{nk} | I_n, \langle \hat{w}_{n\neg k} \rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)] - H(q_w(\cdot; \psi_{nk})) + \text{constant} \\
&= D(q_w(\cdot; \psi_{nk}) \| p_w(w_{nk} | I_n, \langle \hat{w}_{n\neg k} \rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)) + \text{constant}, \qquad \text{(A.11)}
\end{aligned}$$

where $D(\cdot \| \cdot)$ is the Kullback-Leibler divergence. Hence, we can summarize the update rule for the E-step as

$$\hat{\psi}_{nk} \leftarrow \operatorname*{argmin}_{\psi_{nk}} D(q_w(\cdot;\psi_{nk}) \| p_w(\cdot|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)). \tag{A.12}$$

Once we solve (A.12) for each n and k, we can compute all expectations $\langle \hat{w}_{nk}\rangle$ and $\langle \hat{w}_{nk}^2\rangle$. Explicit update equations for $\hat{\psi}_{nk}$, $\langle \hat{w}_{nk}\rangle$, and $\langle \hat{w}_{nk}^2\rangle$ depend on what distributions p_w and q_w are. We derive updates for the case when $p_w(\cdot;\lambda_k)$ is exponential and $q_w(\cdot;\psi_{nk})$ is positive normal in Appendix B, where observing that $p_w(w_{nk}|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$ is positive normal when prior $p_w(\cdot;\lambda_k)$ is exponential eases the calculation.

■ A.2 M-step: Updating Deformations $\boldsymbol{\Phi}$

We also use coordinate ascent for the M-step, beginning with optimizing over deformations $\boldsymbol{\Phi}$. Our derivations here actually do not guarantee coordinate ascent with respect to $\boldsymbol{\Phi}$. Specifically, we first maximize over an approximate lower bound on the variational lower bound $\mathcal{L}^-$ with respect to each Φ_n while ignoring the average deformation constraint, and then project our solution onto a space in which the average deformation constraint is met.

Our derivation for updating each deformation relies on approximation (2.24). Combining this approximation with with variational lower bound (A.5) and dropping terms that do not involve Φ_n, we see that maximizing $\mathcal{L}^-$ with respect to Φ_n is equivalent to minimizing energy functional

$$\begin{aligned}\widetilde{E}(\Phi_n) = \sum_{x\in\Omega} \frac{|\mathbf{J}_{\Phi_n}(x)|}{2\hat{\sigma}^2}\left[\left(I_n(\Phi_n(x)) - \sum_{k=1}^{K}\langle \hat{w}_{nk}\rangle \hat{D}_k(x)\right)^2 + \sum_{k=1}^{K}(\langle \hat{w}_{nk}^2\rangle - \langle \hat{w}_{nk}\rangle^2)\hat{D}_k^2(x)\right] \\ - \log p_\Phi(\Phi_n).\end{aligned} \tag{A.13}$$

Since $|\mathbf{J}_{\Phi_n}(x)| \leq \phi_{\max}$ for all n and x, and $\|D_k\|_2 \leq 1$ for each k, energy $\widetilde{E}(\Phi_n)$ is bounded above by

$$E^+(\Phi_n) = \frac{\phi_{\max}}{2\hat{\sigma}^2}\left[\left\| I_n \circ \Phi_n - \sum_{k=1}^{K}\langle \hat{w}_{nk}\rangle \hat{D}_k\right\|_2^2 + \sum_{k=1}^{K}(\langle \hat{w}_{nk}^2\rangle - \langle \hat{w}_{nk}\rangle^2)\right] - \log p_\Phi(\Phi_n). \tag{A.14}$$

We opt to solve

$$\widetilde{\Phi}_n \leftarrow \operatorname*{argmin}_{\Phi_n} E^+(\Phi_n) = \operatorname*{argmin}_{\Phi_n} \left\{ \frac{\phi_{\max}}{2\hat{\sigma}^2} \left\| I_n \circ \Phi_n - \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle \hat{D}_k \right\|_2^2 - \log p_\Phi(\Phi_n) \right\}, \tag{A.15}$$

which amounts to registering image $\sqrt{\phi_{\max}} I_n$ to image $\sqrt{\phi_{\max}} \sum_{k=1}^{K} \langle \hat{w}_{nk} \rangle \hat{D}_k$ for energy functional (3.3). Assuming that approximation (2.24) is accurate, optimization (A.15) may not maximize variational lower bound $\mathcal{L}^-$ over Φ_n; however, it maximizes a lower bound on $\mathcal{L}^-$ with respect to Φ_n.

Lastly, we project intermediate deformation estimates $\widetilde{\Phi}_n$ to a space where the average deformation is approximately identity. Assuming that each deformation $\Phi_n = \exp(\mathcal{V}_n)$ is sufficiently small and has associated velocity field $\mathcal{V}_n$, then the average deformation can be approximated as $\Phi_1 \circ \cdots \circ \Phi_N \approx \exp(\sum_{n=1}^{N} \mathcal{V}_n)$. Thus, the average deformation is approximately identity when the sum of the velocity fields is 0. With this intuition, let $\widetilde{\mathcal{V}}_n$ be the velocity field of intermediate deformation $\widetilde{\Phi}_n$ from optimization (A.15). Then compute

$$\hat{\Phi}_n \leftarrow \exp\left(\widetilde{\mathcal{V}}_n - \frac{1}{N} \sum_{m=1}^{N} \widetilde{\mathcal{V}}_m \right), \tag{A.16}$$

which ensures that average deformation $\hat{\Phi}_1 \circ \cdots \circ \hat{\Phi}_N$ is approximately identity.

■ A.3 M-step: Updating Parameters $\boldsymbol{\lambda}$ and σ^2

To maximize variational lower bound $\mathcal{L}^-$ over λ_k, observe that it suffices to solve

$$\hat{\lambda}_k \leftarrow \operatorname*{argmax}_{\lambda_k} \sum_{n=1}^{N} \mathbb{E}_{\hat{q}_w}[\log p_w(w_{nk}; \lambda_k) | \boldsymbol{I}, \hat{\boldsymbol{\Phi}}, \hat{\boldsymbol{D}}], \tag{A.17}$$

i.e., maximization of an expected log likelihood. If distribution p_w is in the exponential family, then the resulting maximization is essentially maximum likelihood where expected “soft” counts involving w_{nk} are used.

To maximize variational lower bound $\mathcal{L}^-$ over σ^2, a simple calculation shows that setting the derivative with respect to σ^2 to 0 and rearranging terms gives update

$$\hat{\sigma}^2 \leftarrow \frac{1}{N|\Omega|}\sum_{n=1}^{N}\left[\left\|I_n - \sum_{k=1}^{K}\langle\hat{w}_{nk}\rangle(\hat{D}_k \circ \hat{\Phi}_n^{-1})\right\|_2^2 + \sum_{k=1}^{K}(\langle\hat{w}_{nk}^2\rangle - \langle\hat{w}_{nk}\rangle^2)\|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2\right]. \tag{A.18}$$

■ A.4 M-step: Updating Dictionary D

Lastly, we maximize variational lower bound $\mathcal{L}^-$ over each dictionary element D_k. Fixing k, dropping terms from $\mathcal{L}^-$ that don't depend on D_k, and using approximation (2.24), we find that maximizing $\mathcal{L}^-$ with respect to D_k is equivalent to minimizing energy (3.13) with respect to D_k. Rearranging terms in (3.13) yields

$$E(D_k) = \underbrace{E_{\text{smooth},1}(D_k) + E_{\text{smooth},2}(D_k)}_{E_{\text{smooth}}(D_k)} + E_{\text{separable}}(D_k), \tag{A.19}$$

where:

$$E_{\text{smooth},1}(D_k) = \frac{1}{2\hat{\sigma}^2}\sum_{n=1}^{N}\sum_{x\in\Omega}|\mathbf{J}_{\hat{\Phi}_n}(x)|\left[\left(I_n(\hat{\Phi}_n(x)) - \sum_{k=1}^{K}\langle\hat{w}_{nk}\rangle D_k(x)\right)^2 + \sum_{k=1}^{K}(\langle\hat{w}_{nk}^2\rangle - \langle\hat{w}_{nk}\rangle^2)D_k^2(x)\right], \tag{A.20}$$

$$E_{\text{smooth},2}(D_k) = \frac{\beta}{2}D_k^\top \mathbf{L} D_k, \tag{A.21}$$

$$E_{\text{separable}}(D_k) = \sum_{x\in\Omega}|D_k(x)|\left(\alpha + \gamma\sum_{\ell\neq k}|\hat{D}_\ell(x)|\right). \tag{A.22}$$

We want to minimize (A.19) subject to $\|D_k\|_2 \leq 1$ and there existing ellipsoid $\mathcal{E}_k \in \mathcal{E}(V_{\max}, r_{\max})$ containing the spatial support of D_k.

Convex relaxation: Ignoring the ellipsoid constraint, we're left with a convex program, which is similar to Lasso except that each voxel/index x of D_k has a different regularization parameter $\alpha + \gamma\sum_{\ell\neq k}|\hat{D}_\ell(x)|$, as indicated in eq. (A.22), and, furthermore, we have constraint $\|D_k\|_2 \leq 1$.

As the notation suggests, $E_{\text{smooth},1}(D_k)$ and $E_{\text{smooth},2}(D_k)$ are smooth functions whereas $E_{\text{separable}}(D_k)$ is separable but not smooth. With this decomposition, minimizing (A.19) subject to $\|D_k\|_2 \leq 1$ using fast iterative shrinkage-thresholding [3] is straightforward; the algorithm for the Lasso is in [3] and the only modifications are having the shrinkage operator threshold vary per voxel and projecting onto the ℓ_2 disk at each step. What remains is choosing an appropriate step size so that we don't have to use backtracking.

Following derivations from [3], it suffices to set the step size to $\delta = 1/L$, where L is a Lipschitz constant of $\nabla E_{\text{smooth}}(D_k) = \nabla E_{\text{smooth},1}(D_k) + \nabla E_{\text{smooth},2}(D_k)$. A simple calculation shows that $\nabla E_{\text{smooth},1}(D_k)$ and $\nabla E_{\text{smooth},2}(D_k)$ have Lipschitz constants $\frac{\phi_{\max}}{\hat{\sigma}^2}\sum_{n=1}^N \langle \hat{w}_{nk}^2 \rangle$ and $\beta\|\mathbf{L}\|_2$ respectively:

$$\begin{aligned}
\|\nabla E_{\text{smooth},1}(I) - \nabla E_{\text{smooth},1}(J)\|_2^2 &= \sum_{x\in\Omega}\left(\frac{\partial E_{\text{smooth},1}(I)}{\partial D_k(x)} - \frac{\partial E_{\text{smooth},1}(J)}{\partial D_k(x)}\right)^2 \\
&= \sum_{x\in\Omega}\left[\frac{1}{\sigma^2}\sum_{n=1}^N |\mathbf{J}_{\Phi_n}(x)|\langle w_{nk}^2\rangle(I(x)-J(x))\right]^2 \\
&\le \sum_{x\in\Omega}\left[\frac{\phi_{\max}}{\sigma^2}\sum_{n=1}^N \langle w_{nk}^2\rangle(I(x)-J(x))\right]^2 \\
&= \underbrace{\left[\frac{\phi_{\max}}{\sigma^2}\sum_{n=1}^N \langle w_{nk}^2\rangle\right]^2}_{\text{squared Lipschitz constant}} \underbrace{\sum_{x\in\Omega}(I(x)-J(x))^2}_{\|I-J\|_2^2}, \qquad \text{(A.23)}\\
\|\nabla E_{\text{smooth},2}(I) - \nabla E_{\text{smooth},2}(J)\|_2 &= \|\beta\mathbf{L}I - \beta\mathbf{L}J\|_2 \\
&= \|\beta\mathbf{L}(I-J)\|_2 \\
&\le \underbrace{\beta\|\mathbf{L}\|_2}_{\text{Lipschitz constant}} \|I-J\|_2. \qquad \text{(A.24)}
\end{aligned}$$

Hence, $\nabla E_{\text{smooth}}(D_k)$ has Lipschitz constant

$$L = \frac{\phi_{\max}}{\hat{\sigma}^2}\sum_{n=1}^N \langle \hat{w}_{nk}^2\rangle + \beta\|\mathbf{L}\|_2. \qquad \text{(A.25)}$$

Of course, $\nabla E_{\text{smooth}}(D_k)$ is also Lipschitz continuous with any constant larger than L, so we could upper bound $\|\mathbf{L}\|_2$ and effectively use a smaller step size δ.

Rounding to enforce the ellipsoid constraint: Here, we just fill in a few details for the rounding procedure in Section 3.2. First, note that the intensity "mass" image is given by

$$M(c) = \sum_{x \in \mathcal{B}_c} |\widetilde{D}_k(x)|^2 = \sum_{x \in \mathcal{B}_c} \widetilde{D}_k^2(x) = \sum_{x} \widetilde{D}_k^2(x) B(c-x), \tag{A.26}$$

where B is the image associated with a ball of radius $r_{\max}$ centered at the origin:

$$B(x) \triangleq \begin{cases} 1 & \text{if } \|x\|_2 \leq r_{\max}, x \in \Omega, \\ 0 & \text{if } \|x\|_2 > r_{\max}, x \in \Omega. \end{cases} \tag{A.27}$$

Without loss of generality, we can assume Ω contains the spatial support of image B; otherwise, we just need to zero-pad or shift coordinates. Next, recognizing that eq. (A.26) is a convolution, we compute M in the frequency domain: $M \leftarrow \mathcal{F}^{-1}\{\mathcal{F}\{D^2\} \odot \mathcal{F}\{B\}\}$, where $\mathcal{F}$ is the multi-dimensional discrete Fourier transform.

We conclude this section by elaborating on why the ellipsoid fitting for Alg. 5 is approximate. Specifically, the problem can be rephrased as trying to preserve as much squared ℓ_2 norm of an input image I as possible when masking it with an ellipsoid of volume $V_{\max}$, where we assume that we've already zeroed out all elements of I except for within spatial support defined by some ball of radius $r_{\max}$. Without loss of generality, we can assume this ball to be centered at the origin. Thus, we seek a solution to

$$\max_{\text{ellipsoid } \mathcal{E}} \sum_{x \in \mathcal{E}} I^2(x) \quad \text{subject to:} \quad \text{vol}(\mathcal{E}) = V_{\max}. \tag{A.28}$$

With parameterization $\mathcal{E} = \{x \in \Omega : (x-v)^\top \mathbf{A}(x-v) \leq 1\}$ where $\mathbf{A}$ is positive semidefinite (denoted $\mathbf{A} \succeq 0$) and voxel v is in the convex hull Ω_c of Ω, then the above optimization problem can be rewritten as

$$\begin{aligned} \max_{\mathbf{A} \succeq 0, v \in \Omega_c} & \sum_{x \in \Omega} I^2(x) \mathbf{1}\{(x-v)^\top \mathbf{A}(x-v) \leq 1\} \\ \text{subject to:} & \quad \log\det \mathbf{A} = \kappa, \end{aligned} \tag{A.29}$$

where constant κ ensures that the volume of the ellipsoid, which scales with $\det \mathbf{A}$, is $V_{\max}$. The crux of the problem is assigning which voxels are "outliers" that should not be in the ellipsoid and which voxels should be in the ellipsoid. Rather than maximizing the nonconcave objective in (A.29), we maximize a lower bound on the objective by noting that

$$\mathbf{1}\{(x-v)^\top \mathbf{A}(x-v) \leq 1\} \geq 1 - (x-v)^\top \mathbf{A}(x-v), \tag{A.30}$$

which follows from $\mathbf{A}$ being positive semidefinite. Thus, we instead solve

$$\max_{\mathbf{A}\succeq 0, v\in\Omega_c} \sum_{x\in\Omega} I^2(x)(1-(x-v)^\top \mathbf{A}(x-v))$$
$$\text{subject to:} \quad \log\det\mathbf{A} = \kappa. \tag{A.31}$$

This problem is nearly identical to maximum likelihood estimation for a multivariate Gaussian! The derivation is nearly identical as well, so we just state the solution:

$$v = \sum_{x\in\Omega} \frac{I^2(x)}{\sum_{y\in\Omega} I^2(y)} x, \tag{A.32}$$

$$\mathbf{A}^{-1} \propto \sum_{x\in\Omega} I^2(x)(x-v)(x-v)^\top, \tag{A.33}$$

where the constant of proportionality ensures that the ellipsoid has volume $V_{\max}$. This ellipsoid fit can be interpreted as just fitting a Gaussian to a distribution proportional to $I^2(\cdot)$ and then rescaling the ellipsoid defining the Gaussian's covariance to have volume $V_{\max}$.

Appendix B

Deriving the Specialized Inference Algorithm for Chapter 4

We consider the case where $p_w(w_{nk};\lambda_k) = \lambda_k e^{-\lambda_k w_{nk}}$ with $w_{nk} \geq 0$ and $\lambda_k > 0$, and $q_w(w_{nk};\psi_{nk}) = \mathcal{N}^+(w_{nk};\mu_{nk},\nu_{nk})$, i.e., $\psi_{nk} = (\mu_{nk},\nu_{nk})$. It suffices to derive updates for estimating variational distribution parameters $\boldsymbol{\psi}$ (optimization problem (A.12) in the E-step) and estimating prior distribution parameters $\boldsymbol{\lambda}$ (optimization problem (A.17) from the M-step).

Estimating variational distribution parameters $\boldsymbol{\psi}$: We begin by showing that posterior $p_w(w_{nk}|I_n, \langle\hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$ in (A.10) is positive normal. Then we solve the optimization problem in update rule (A.12) by setting the KL divergence to 0, which effectively means finding the parameters of posterior $p_w(w_{nk}|I_n, \langle\hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$. We shall use the fact that the positive normal density $\mathcal{N}^+(\cdot;\mu,\nu)$ can be written in exponential family form:

$$\begin{aligned}\mathcal{N}^+(w;\mu,\nu) &= \exp\left\{\frac{\mu}{\nu}w - \frac{1}{2\nu}w^2 - \frac{\mu}{2\nu} - \frac{1}{2}\log(2\pi\nu) - \log Q\left(-\frac{\mu}{\sqrt{\nu}}\right)\right\} \\ &\propto \exp\left\{\frac{\mu}{\nu}w - \frac{1}{2\nu}w^2\right\},\end{aligned} \tag{B.1}$$

where $w \geq 0$ and $Q(s) \triangleq \int_s^\infty \frac{1}{\sqrt{2\pi}}e^{-t^2/2}dt$ is the tail probability of the standard normal distribution. Specifically, the natural parameters of the positive normal distribution in exponential family form (B.1) are $(\frac{\mu}{\nu}, -\frac{1}{2\nu})$ and the natural statistics are (w, w^2).

With proportionality $p_w(w_{nk}; \lambda_k) \propto \exp\{-\lambda_k w_{nk}\}$ and eq. (A.10), we have

$$\begin{aligned}
&\log p_w(w_{nk}|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2) \\
&= \left[\frac{1}{\hat{\sigma}^2} \sum_{x\in\Omega} \left(I_n(x) - \sum_{\ell\neq k} \langle \hat{w}_{n\ell}\rangle \hat{D}_\ell(\hat{\Phi}_n^{-1}(x)) \right) \hat{D}_k(\hat{\Phi}_n^{-1}(x)) - \hat{\lambda}_k \right] w_{nk} \\
&\quad - \left[\frac{1}{2\hat{\sigma}^2} \sum_{x\in\Omega} \hat{D}_k^2(\hat{\Phi}_n^{-1}(x)) \right] w_{nk}^2 + \text{constant},
\end{aligned} \tag{B.2}$$

where we must have $w_{nk} \geq 0$ as stipulated by the exponential prior on w_{nk}. This density directly corresponds to the log of exponential family form (B.1) of a positive normal, so we conclude that posterior $p_w(w_{nk}|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$ is positive normal.

Next, we minimize

$$D(q_w(\cdot;\psi_{nk}) \| p_w(\cdot|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2), \tag{B.3}$$

i.e., we solve (A.12). First, observe that $q_w(\cdot;\psi_{nk})$ is positive normal, so using form (B.1), we have:

$$\log q_w(w_{nk}; \mu_{nk}, \nu_{nk}) = \frac{\mu_{nk}}{\nu_{nk}} w_{nk} - \frac{1}{2\nu_{nk}} w_{nk}^2 + \text{constant}. \tag{B.4}$$

We are now ready for the key step: The KL divergence between the two distributions is 0 (i.e., minimized) when the two distributions are the same, which happens if we match the *natural parameters* of the (log) exponential family forms of posterior $p_w(w_{nk}|I_n, \langle \hat{w}_{n\neg k}\rangle, \hat{\Phi}_n, \hat{\boldsymbol{D}}; \hat{\lambda}_k, \hat{\sigma}^2)$ and approximating distribution $q_w(w_{nk}; \mu_{nk}, \nu_{nk})$, i.e., match the natural parameters from eqs. (B.2) and (B.4):

$$\frac{\mu_{nk}}{\nu_{nk}} = \frac{1}{\hat{\sigma}^2} \sum_{x\in\Omega} \left[\left(I_n(x) - \sum_{\ell\neq k} \langle \hat{w}_{n\ell}\rangle \hat{D}_\ell(\hat{\Phi}_n^{-1}(x)) \right) \hat{D}_k(\hat{\Phi}_n^{-1}(x)) \right] - \hat{\lambda}_k, \tag{B.5}$$

$$-\frac{1}{2\nu_{nk}} = -\frac{1}{2\hat{\sigma}^2} \sum_{x\in\Omega} \hat{D}_k^2(\hat{\Phi}_n^{-1}(x)). \tag{B.6}$$

Solving for μ_{nk} and ν_{nk}, we obtain the update rules:

$$\hat{\mu}_{nk} \leftarrow \frac{\langle I_n - \sum_{\ell\neq k} \langle \hat{w}_{n\ell}\rangle (\hat{D}_\ell \circ \hat{\Phi}_n^{-1}), \hat{D}_k \circ \hat{\Phi}_n^{-1} \rangle - \hat{\sigma}^2 \hat{\lambda}_k}{\|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2}, \tag{B.7}$$

$$\hat{\nu}_{nk} \leftarrow \frac{\hat{\sigma}^2}{\|\hat{D}_k \circ \hat{\Phi}_n^{-1}\|_2^2}. \tag{B.8}$$

Next, computing $\langle \hat{w}_{nk} \rangle$ and $\langle \hat{w}_{nk}^2 \rangle$ is straightforward since distribution $w \sim \mathcal{N}^+(\mu, \nu)$ has first and second moments:

$$\mathbb{E}[w] = \mu + \frac{\sqrt{\nu}\exp(-\mu^2/(2\nu))}{\sqrt{2\pi}Q(-\mu/\sqrt{\nu})}, \tag{B.9}$$

$$\mathbb{E}[w^2] = \nu + \mu^2 + \frac{\mu\sqrt{\nu}\exp(-\mu^2/(2\nu))}{\sqrt{2\pi}Q(-\mu/\sqrt{\nu})}. \tag{B.10}$$

Estimating prior parameters $\boldsymbol{\lambda}$*:* To update λ_k, we solve solving optimization problem (A.17) from the M-step. With $p_w(w_{nk}; \lambda_k) = \lambda_k e^{-\lambda_k w_{nk}}$, the problem becomes

$$\hat{\lambda}_k \leftarrow \underset{\lambda_k}{\operatorname{argmax}} \left\{ -\lambda_k \sum_{n=1}^{N} \langle \hat{w}_{nk} \rangle + N \log \lambda_k \right\}, \tag{B.11}$$

which has a derivation identical to that of maximum likelihood for the exponential distribution where each "observed" count is instead an expected count. Thus, we just state the final result:

$$\hat{\lambda}_k \leftarrow \frac{1}{\frac{1}{N}\sum_{n=1}^{N} \langle \hat{w}_{nk} \rangle}. \tag{B.12}$$

Bibliography

[1] K. Amunts, A. Schleicher, U. Bürgel, H. Mohlberg, H. Uylings, and K. Zilles. Broca's region revisited: Cytoarchitecture and intersubject variability. *Journal of Comparative Neurology*, 412(2):319–341, 1999. ISSN 1096-9861.

[2] V. Arsigny, O. Commowick, X. Pennec, and N. Ayache. A log-euclidean framework for statistics on diffeomorphisms. In *Proc. Medical Image Computing and Computer-Assisted Intervention*, volume 4190 of *LNCS*, pages 924–931, 2006.

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183, 2009.

[4] M. Brett, I. S. Johnsrude, and A. M. Owen. The problem of functional localization in the human brain. *Nature*, 2002.

[5] P. Cachier and N. Ayache. Isotropic energies, filters and splines for vector field regularization. *Journal of Mathematical Imaging and Vision*, 20(3):251–265, 2004. ISSN 0924-9907.

[6] P. Cachier, E. Bardinet, D. Dormont, X. Pennec, and N. Ayache. Iconic feature based nonrigid registration: the pasha algorithm. *Computer Vision and Image Understanding*, 89(2-3):272–298, 2003.

[7] D. L. Collins, P. Neelin, T. M. Peters, and A. C. Evans. Automatic 3d intersubject registration of mr volumetric data in standardized talairach space. *Journal of Computer Assisted Tomography*, 18(2):192–205, 1994.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1): 1–38, 1977.

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

[10] F. Dru and T. Vercauteren. An itk implementation of the symmetric log-domain diffeomorphic demons algorithm. Insight Journal – 2009 January - June, May 2009.

[11] S. Durrleman, M. Prastawa, G. Gerig, and S. Joshi. Optimal data-driven sparse parameterization of diffeomorphisms for population analysis. In *Proc. Information Processing in Medical Imaging*, volume 22, pages 123–34, 07 2011.

[12] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006.

[13] A. C. Evans, D. L. Collins, S. R. Mills, E D Brown, R L Kelly, and Terry M. Peters. 3D statistical neuroanatomical models from 305 MRI volumes. *Proceedings of the IEEE Nuclear Science Symposium and Medical Imaging Conference*, 3(1-3): 1813–1817, 1993.

[14] E. Fedorenko, P-J Hsieh, A. Nieto-Castañón, S. Whitfield-Gabrieli, and N. Kanwisher. New method for fMRI investigations of language: Defining ROIs functionally in individual subjects. *Neurophysiology*, 104:1177–1194, 2010.

[15] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61:55–79, 2005.

[16] K. J. Friston, J. T. Ashburner, S. J. Kiebel, T. E. Nichols, and W. D. Penny, editors. *Statistical Parametric Mapping: The Analysis of Functional Brain Images*. Academic Press, 2007.

[17] O. P. Hinds, N. Rajendran, J. R. Polimeni, J. C. Augustinack, G. Wiggins, L. L. Wald, H. D. Rosas, A. Potthast, E. L. Schwartz, and B. Fischl. Accurate prediction of v1 location from cortical folds in a surface coordinate system. *Neuroimage*, 39 (4):1585–1599, 02 2008.

[18] A. K. Jain and D. Zongker. Representation and recognition of handwritten digits using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1386–1390, 1997.

[19] A. Klein, S. S. Ghosh, B. Avants, B.T.T. Yeo, B. Fischl, B. Ardekani, J. C. Gee, J.J. Mann, and R. V. Parsey. Evaluation of volume-based and surface-based brain image registration methods. *NeuroImage*, 51(1):214 – 220, 2010.

[20] S. Kurtek, A. Srivastava, and W. Wu. Signal estimation under random time-warpings and nonlinear signal alignment. In *NIPS*, pages 675–683, 2011.

[21] D. Lashkari. *In Search of Functional Specificity in the Brain: Generative Models for Group fMRI Data.* PhD thesis, Massachusetts Institute of Technology, June 2011.

[22] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 801–808. MIT Press, Cambridge, MA, 2007.

[23] M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2):337–365, 2000.

[24] Xiaoguang Lu and Anil K. Jain. Deformation analysis for 3d face matching. In *Proc. Seventh IEEE Workshops on Application of Computer Vision*, WACV-MOTION '05, pages 99–104, 2005.

[25] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Proc. IEEE International Conference on Computer Vision*, pages 2272–2279, 2009.

[26] S. Ogawa, T-M Lee, A. S. Nayak, and P. Glynn. Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields. *Magnetic Resonance in Medicine*, 14(1):68–78, 1990.

[27] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[28] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: a strategy employed by v1. *Vision Research*, 37:3311–3325, 1997.

[29] P. Risholm, S. Pieper, E. Samset, and W. M. Wells. Summarizing and visualizing uncertainty in non-rigid registration. In *Proc. Medical Image Computing and Computer-Assisted Intervention*, volume 6362 of *LNCS*, pages 554–561, 2010.

[30] C. S. Roy and C. S. Sherrington. On the regulation of the blood-supply of the brain. *Journal of Physiology (London)*, 11:85–108, 1890.

[31] M. R. Sabuncu, B. D. Singer, B. Conroy, R. E. Bryan, P. J. Ramadge, and J. V. Haxby. Function-based intersubject alignment of human cortical anatomy. *Cerebral Cortex*, 20(1):130–140, 2010.

[32] J. Talairach and P. Tournoux. *Co-Planar Stereotaxic Atlas of the Human Brain: 3-Dimensional Proportional System: An Approach to Cerebral Imaging*, volume 39. Thieme, 1988.

[33] B. Thirion, P. Pinel, S. Mériaux, A. Roche, S. Dehaene, and J-B Poline. Analysis of a large fMRI cohort: Statistical and methodological issues for group analyses. *NeuroImage*, 35(1):105–120, 2007.

[34] B. Thirion, P. Pinel, A. Tucholka, A. Roche, P. Ciuciu, J-F Mangin, and J-B Poline. Structural analysis of fMRI data revisited: improving the sensitivity and reliability of fMRI group studies. *IEEE Transactions in Medical Imaging*, 26(9): 1256–1269, 2007.

[35] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)*, 58:267–288, 1996.

[36] T. Vercauteren, X. Pennec, A. Perchant, and N. Ayache. Non-parametric diffeomorphic image registration with the demons algorithm. In *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, volume 4792 of *LNCS*, pages 319–326, 2007.

[37] T. Vercauteren, X. Pennec, A. Perchant, and N. Ayache. Symmetric log-domain diffeomorphic registration: A demons-based approach. In *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, volume 5241 of *LNCS*, pages 754–761, 2008.

[38] M. J. Wainwright and M. I. Jordan. *Graphical Models, Exponential Families, and Variational Inference.* Now Publishers Inc., Hanover, MA, USA, 2008. ISBN 1601981848, 9781601981844.

[39] L. Xu, T. D. Johnson, T. E. Nichols, and D. E. Nee. Modeling inter-subject variability in fMRI activation location: a bayesian hierarchical spatial model. *Biometrics*, 65(4):1041–1051, 2009.

[40] J. Yang, K. Yu, Y. Gong, and T. S. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1794–1801. IEEE, 2009.

[41] X-D Zhang. Two sharp upper bounds for the laplacian eigenvalues. *Linear Algebra and its Applications*, 376(0):207 – 213, 2004.