Feature Identification

by

Justin Shaw

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
of
Applied Mathematics

Waterloo, Ontario, Canada, 2020

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

Supervisor:	Marek Stastna Professor, Dept. of Applied Mathematics, University of Waterloo
Internal Member:	Matthew Scott Associate Professor, Dept. of Applied Mathematics, University of Waterloo
Internal Member:	Michael Waite Associate Professor, Dept. of Applied Mathematics, University of Waterloo
Internal-External Member:	Andrea Scott Assistant Professor, Dept. of Systems Design Engineering, University of Waterloo
External Examiner:	Christopher Essex Professor, Dept. of Applied Mathematics, University of Western Ontario

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

We present several methods for identifying time periods of interest (features) in a wide range of data sets.

The gamma method is a computationally inexpensive, flexible feature identification method which uses a comparison of time series to identify a rank-ordered set of features in geophysically-sourced data sets. Many physical phenomena perturb multiple physical variables nearly simultaneously, and so features are identified as time periods in which there are local maxima of absolute deviation in all time series. Unlike other available methods, this method allows the analyst to tune the method using their knowledge of the physical context. The method is applied to a data set from a moored array of instruments deployed in the coastal environment of Monterey Bay, California, and a data set from sensors placed within the submerged Yax Chen Cave System in Tulum, Quintana Roo, Mexico. These example data sets demonstrate that the method allows for the automated identification of features which are worthy of further study. The gamma method appeared as [52].

The EOF error map method is a feature identification method for time-indexed model output. The method is used as a diagnostic to quickly focus the attention on a subset of the data before further analysis methods are applied. Mathematically, the infinity norm errors of empirical orthogonal function (EOF) reconstructions are calculated for each time output. The result is an EOF reconstruction error map which clearly identifies features as changes in the error structure over time. The ubiquity of EOF-type methods in a wide range of disciplines reduces barriers to comprehension and implementation of the method. We apply the error map method to three different Computational Fluid Dynamics (CFD) data sets as examples: the development of a spontaneous instability in a large amplitude internal solitary wave, an internal wave interacting with a density profile change, and the collision of two waves of different vertical mode. In all cases the EOF error map method identifies relevant features which are worthy of further study. The EOF error map method appeared as [51]. Together, the gamma and EOF error map methods allow feature identification in an extremely wide variety of data sets.

While the associated methods papers required brevity and specificity, the thesis is written from the perspective of the overarching research program. This thesis expands the twenty pages or so of material in [52] and [51] to a detailed, over 100 page account of how and why the methods were developed. It includes a much more comprehensive framing of the general problem both methods solve, much more motivation, discussion,

and mathematical background, an entire section on ensemble data sets, including another method for feature identification, examples of the methods applied to full scale data sets, and an appendix of related work. This is the definitive guide to our methodology and results.

Acknowledgements

Thank you to Ryan K. Walter and Aaron Coutino for their assistance with the gamma method. Thanks to Eduard Reinhardt for access to the cenote data sets and feedback on the gamma method. We acknowledge S. Monismith (Stanford) and B. Woodson (University of Georgia) for their help in the collection and original analysis of the Monterey Bay data set ([65]). Thanks to Susan Allen (UBC) for suggestions. Thank you to Ben Storer and Chelsi McNeill for assistance with gamma method tutorial codes. Thanks to Andrew Grace for assistance in editing. Thank you to Ed Vrscay for his assistance, especially as it relates to SSIM. We would also like to thank the University of Waterloo Water Institute for facilitating travel. Thank you to NSERC for their continual support. Thank you to Marek Stastna for his supervision, patience, and kindness. Finally, a special thanks to the committee for their suggestions and support: Matt Scott, Mike Waite, Andrea Scott, and Chris Essex.

Dedication

To all who have supported me.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Preface

Very few theses are read by anyone outside the PhD defence process. Perhaps some lab or family members will take interest, but generally the thesis is an exercise for the candidate, and evidence for the committee. But evidence of what?

Applied Mathematics includes a wide variety of subdisciplines. Coming from a Pure Mathematics undergraduate degree, I had expected to find Applied Mathematics to be as applied as Pure Mathematics was pure. This is not the case. Some Applied Mathematics is "applied" because it was inspired by a physical problem. Some is applied because it could be applied in theory. I struggled to codify exactly what I meant by Applied Mathematics, until one day while studying sabermetrics, I came upon an essay by Keith Woolner entitled"Baseball's Hilbert problems," originally published in [27]. The article discussed how their research program should be directed in order for it to be useful:

"To be relevant, sabermetrics must inform a decision."

This is how I think about Applied Mathematics, and is the standard I applied to my own research program. Ideally, I wanted my mathematics to inform decisions by non-Mathematicians. As Woolner points out, "... in order to produce useful information, you have to start with a relevant question that needs answering." I needed a relevant question from a non-Mathematician whose answer would inform their decisions.

I had not yet solidified this viewpoint in the first few months of my PhD. We knew that we wanted to do some data analysis on common geophysical data sets. Ryan K. Walter at Cal Poly San Luis Obispo provided a data set from Monterey Bay California, and we

set about applying our fledgling techniques to this data set. This included a great deal of mathematical due diligence, which helped form our own foundational understanding of the subject. However, when we showed him our results, they were not very useful to him. We asked him what he would want out of a data analysis method. He replied "Is there a way to identify epochs within data sets?" To paraphrase,

"Is there a way to mathematically identify interesting times in a data set?"

Answering this question helps him detect what he's interested in: physical events and processes such as waves, storms, and the like.

I did not know it at the time, but this was the question I needed: a relevant question from a non-Mathematician whose answer would inform decisions during data analysis. The initial answer to Ryan's question led to the gamma method, useful for many typical data sets gathered from the field. A second method more useful on other common data sets, such as those generated by computational models, was developed next. Together, these two methods allow us to find interesting times in a wide variety of data sets, including those gathered from fluid dynamics, but also many many more. These methods have been effective in every context to which they have been applied. In essence, this entire document is an extensive, detailed answer to Ryan's original question, and one that has been well-received and employed by non-Mathematicians to inform decisions. In the Applied Mathematics sense just defined, this means the PhD has been an overwhelming success.

1.2 Overview

Logistical support is tremendously undervalued. The end goal of an endeavour is often the only part which is evident to the wider world. As a result, those who made the work possible are invisible to most people, and so few aspire to support valuable work. There is a perception that those who facilitate others are less valuable than those they support. But who can work without food, sleep, a workspace, supplies, tools, companionship, guidance, and an end goal of value? This is of course general, but in particular logistical support is what makes academic endeavours possible. So while we may stand on the shoulders of giants to make academic discoveries, we would have nowhere to stand at all without the continual support of an enormous network of people who generally get no credit for their part in our victories.

In this thesis, we undertake the task of supporting those who have more data than they know what to do with. This is nearly every modern academic who is not themselves a

data analyst. While former generations of scientists struggled to find data, our generation's challenge is to sort through the massive data sets that we have acquired. The question is not "how can we get a data set?" but rather "which parts of this data set are relevant?" Put another way, we might ask "of all the facts we know, which are the most important?" We will see that the methods developed in this thesis to answer questions of this type apply to an extremely wide range of data sets.

The methods we have written are extremely useful, but they will not be employed if they are not understood. This thesis had to be written so that non-mathematicans can and will read it. As a mathematics thesis it must also have the appropriate specificity and rigour. To satisfy all requirements, we have adopted some presentation conventions we will now outline. We are geophysical fluid dynamicists, and so we will use examples from our field to illustrate our points, but the subject of this thesis is data analysis, not fluid dynamics. This being the case, in every example we will introduce only enough context as is required for the exposition of the data analysis. This will require a small amount of specialist language, which will be defined in brackets or given a reference as appropriate. Otherwise all physical details of the datasets which do not assist with the exposition of the data analytic methods are omitted. This means that generally data sets are presented in terms of grid points, time output number, and numeric field values, with units only being provided for context where necessary. This makes the thesis much more accessible, as we avoid unnecessary digressions and definitions. It should also be noted that every Figure in the thesis was generated in our lab, and all data sets are ours, unless otherwise referenced.

Before you employ these methods for yourself, you might be interested in why these methods are useful to us. Figure 1.1 shows a single time output from one of our high resolution simulations. At the time, we had not yet developed the methods outlined in this thesis. As a result, in order to focus our analysis efforts we had to go through the large, unwieldy data set manually. This required us to consider many different time outputs, as well as spend a few days visualizing those outputs before we settled on this time output. The methods presented here could have replaced all of that fumbling with a mathematical choice of output for the price of a small amount of CPU time. After the output was chosen, more time was spent tweaking the visualization, until the final image was produced. This image was featured as part of a visualization showcase at the High-Performance Computing Symposium in Edmonton in June of 2016, and was also included in Compute Canada's annual report as an example of research being enabled by their systems. Knowing what we know now, we could have accomplished the same result in a shorter time and in a more objective way.

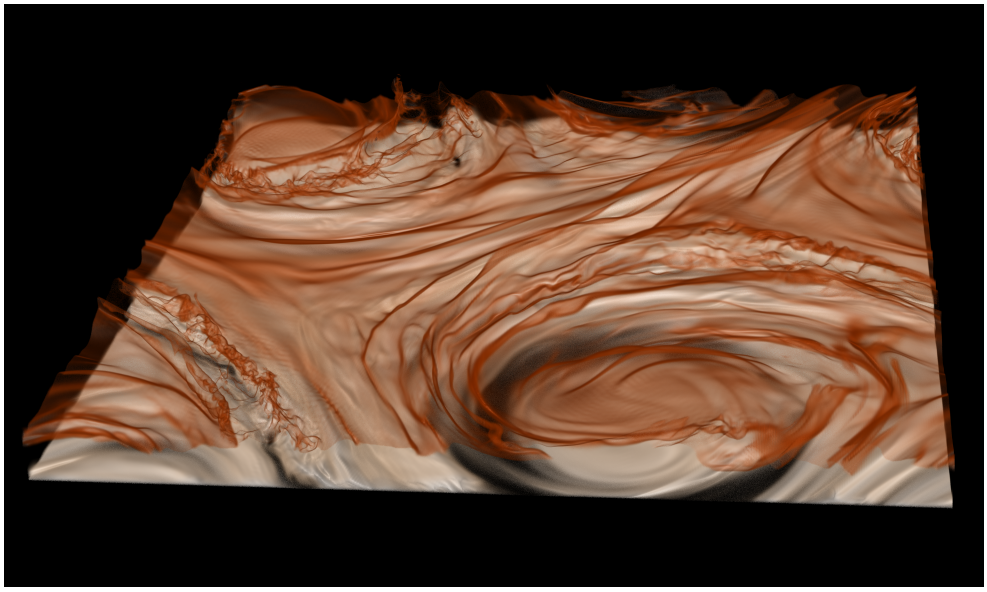

Figure 1.1: A three-dimensional density field is shown. A jet of water injected into the side of a stratified fluid causes the formation of large vortices (in cream white), and associated small scale filaments (in sand red).

As we will outline in section 4.2, many tools have been developed to study structures such as those in Figure 1.1. To be clear, that is not the purpose of this thesis. Rather than building tools to analyse particular physical phenomena in a narrow range of data sets, we have instead built tools to find time periods of interest in a broad range of data sets. We are not studying particular phenomena, we are finding time periods which are worth a closer look. There are already an enormous number of existing analysis methods for data sets as we will discuss in sections 2.2 and 4.2. However, we know of no other methods for mathematically identifying time periods of interest aside from those we have devised and presented here.

Time periods of interest will be referred to as "features" of the data set. This is a deliberately vague term meant to convey the idea that something is happening in the data set, but further study by the analyst is required. This definition requires that the data set have a time dimension, which we will assume throughout this thesis unless otherwise stated. The choice to focus on time-indexed data means we are no longer considering the completely general problem of finding the important part in any data set at all. This is still a fairly weak assumption, as many data sets are in this category, including the

output of every dynamic model. As we will discuss, it may be possible to extend these methods to data sets without a time dimension in certain contexts. However there are a great many types of data sets and it is unlikely that a single analysis method (or two or three) could be constructed which would work perfectly in every context imaginable.

That said, the following methods have been successful in every context to which they has been applied.

Chapter 2

The Gamma Method

2.1 Author's Note

This chapter (2) originally appeared as [52]. The presentation here is slightly expanded mostly to replace the terse language of publication with the more explanatory prose we have adopted for the thesis.

2.2 Introduction

Geophysical researchers often study physical phenomena using instrument arrays sampling the physical variables affected by those phenomena at multiple spatial locations. This produces a data set consisting of vector time series. Features in the data set are often identified by methods such as the visual inspection of plots, or other ad hoc means. As the size and quality of geophysically-sourced time series data sets increase these methods become labor-intensive. Automated methods of identifying a set of features worthy of further study are needed.

There are an enormous variety of vector time series analysis techniques available. Empirical Orthogonal Functions (EOF) [17]; more general dimension-reduction type methods [43]; wavelet [63], Fourier, harmonic, and spectral analysis methods [10]; data smashing [5]; similarity measure approaches [70]; data mining techniques [32]; and many more methods of varying mathematical sophistication. However, generally, existing vector time series analysis techniques are developed from a series of mathematical assumptions

and then applied to data sets in a purely mathematical sense, free of physical information except for that encoded as parameters for the method. This abstraction is done both to satisfy the demands of mathematical rigor and to make the method applicable in a wide array of contexts. However, such methods apply in almost every context precisely because they largely ignore changes due to context. In particular it can become very difficult to combine the analyst's knowledge of the physical context with the interpretation of the method's output.

Many methods depend on mathematical information which may be difficult to derive from the known physical context. So for example, some methods require a choice of statistical model in order to draw comparisons [25]. The results of the method depend on the statistical model chosen, but in many geophysical contexts it is not at all clear which model should be used. Moreover many statistical methods only apply to data assumed to be of a certain mathematical form, such as ergodic, steady state, etc. In many geophysical contexts it is not reasonable to adopt such assumptions on the form of the data (see for example [41]). Nonparametric approaches such as [39] avoid mathematical assumptions on the form of the underlying distribution, but still use mathematical tools like cost functions whose effect on the physical interpretation of the method's output can be difficult to determine. Even if certain mathematical assumptions are appropriate in a given context, not all researchers will have the background necessary to encode their knowledge of the physical context in a statistical model. If the researcher does not know what part of the method's output is from the physics, and what part is from the underlying mathematics, their confidence in deriving conclusions about the physics will be severely limited. Finally, for practical purposes, more advanced data analysis methods are often limited in their usefulness by the availability of user-friendly software (e.g., the open and widely used package by [61]). The method we present ameliorates all the concerns just listed, because it uses the researcher's knowledge of the physical context without requiring them to quantify it for use in a mathematical formalism.

One may rebut the concerns just outlined by pointing out that standard methods in geosciences could be used because their physical interpretations have been made clear over time through widespread use. However familiar methods are not well suited to identifying features in vector time series caused by physical phenomena. For example EOF-type methods ([17]) can process such data sets, but the focus here is on identification of events whose time duration is much shorter than the total record. EOFs are variance-maximizing, and while high total variance in a mode may be the result of an event, it may also be the result of low variance over the entire record. Methods of this type are therefore ill-suited for event detection. Similarly methods for comparing two time series abound, e.g. correlation, covariance, or coherence [10]; [61], but when these

methods are applied pairwise to a data set with more than two series there is a combinatorial explosion of options: if there are k series, there are $\binom{k}{2} = k(k-1)/2$ such pairs. There are algorithms that address this issue [36] but the sophistication of the mathematics ramps up quickly. The method presented here can be applied to any number of time series simultaneously, subject only to memory constraints.

The purpose of this chapter is not to downplay the value of existing methods, but rather to present a method for those researchers who would gladly trade some mathematical sophistication for a clearer link with the known physical context and a lower implementation cost. We present a physics-based, computationally inexpensive, flexible, easily-implemented, and transparent method for the automated identification of features caused by physical phenomena. We call this method 'the gamma method,' and it is outlined in section 2.3. In section 2.4 the method is applied to a data set from the coastal environment of Monterey Bay, Cailfornia (2.4.1), and a data set from the Yax Chen Cave System, near Tulum, Mexico (2.4.2). Section 2.5 includes further discussion. The publication includes tutorial codes for the gamma method written in MATLAB, R, and python in the supplementary material.

2.3 Methods

2.3.1 The Gamma Method

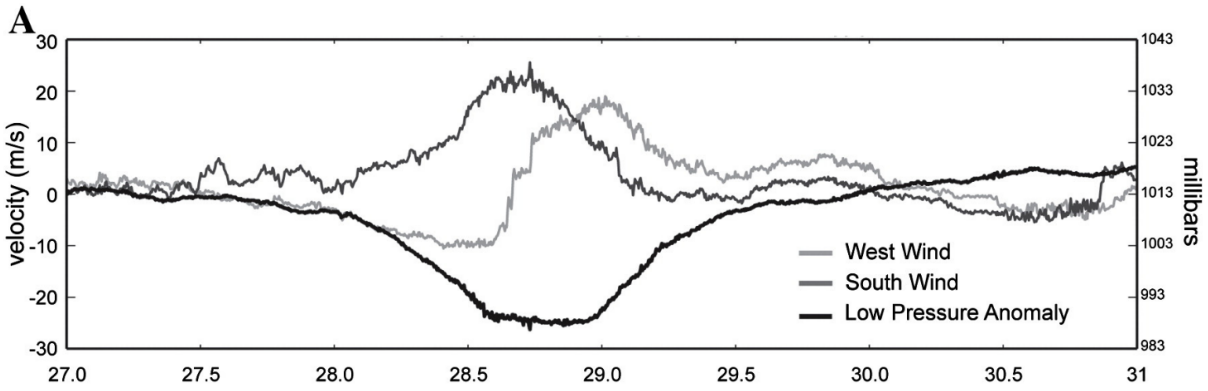

Figure 2.1: Figure 4A from [38]. The time axis is measured in days of August, 2011. As Tropical Storm Irene passes the meteorological station there are clear deviations from the background state of the physical variables.

Before details are presented we outline the gamma method in broad terms. We expect that physical phenomena of interest will impact multiple physical quantities nearly simultaneously. For example, Figure 2.1 reproduces Figure 4A of [38], which shows deviations in wind speeds and air pressure as tropical storm Irene passes a meteorological station. This is an example of the fact that the physical quantities impacted by an event lead to deviations from the background state in the associated time series (wind speed and pressure in this case).

To streamline the presentation we assume that the data has been controlled for quality and filtered by whatever methods the discipline deems appropriate. Assume the data set consists of time series $\{x_1(t), x_2(t), \ldots, x_k(t)\}$ sampling multiple physical quantities with sensors nearby one another, as they would be in a single instrument cluster such as the one used in [38]. Note that time would actually be discrete here, as the time series are formed by sampling at the sensor's rate. We present in continuous time to avoid a second index. We have now formulated the problem:

Problem Statement. *Given a data set consisting of time series* $\{x_1(t), x_2(t), \ldots, x_k(t)\}$*, identify time periods (features) denoted* $\{\mathcal{F}_1, \mathcal{F}_2, \ldots\}$ *in which all* $x_i(t)$ *experience a deviation from their respective trends.*

To solve this problem, we proceed as follows. For each time series $x_i(t)$, form the associated absolute deviation series

$$\hat{x}_i(t) = \kappa_i |x_i(t) - \mu_i(t)| \tag{2.1}$$

where κ_i is a scaling constant and $\mu_i(t)$ is some trend chosen by the analyst as appropriate to the physical context. Large values of $\hat{x}_i$ correspond to large deviations from the trend, and small values correspond to values of x_i near the trend. Absolute deviation rather than standard deviation is used to avoid accentuating outliers. The absolute deviation series is still affected by outliers, but accentuates them less than the corresponding standard deviation series. For an in-depth discussion see [22]. Features in the data set are identified using the maxima of the time series

$$\gamma(t) = \min_i \{\hat{x}_i(t)\} = \min_i \{\kappa_i |x_i(t) - \mu_i(t)|\} \tag{2.2}$$

at every time t (note that $\gamma(t) \geq 0$). We will call the set of time series $\{x_i\}$ included in the definition of $\gamma(t)$ the 'defining set' of time series for $\gamma(t)$. Notice also that by construction of $\gamma(t)$, any number of time series may be in the defining set, so this method is not a pairwise comparison method.

The key observation is this: because $\gamma(t)$ is defined as the minimum curve, if it is perturbed from zero, all curves are perturbed from zero. Therefore, if we wish to find times where all time series are experiencing deviations from their respective trends, we should look for deviations in $\gamma(t)$. In particular, the maxima of $\gamma(t)$ correspond to times when all physical quantities sampled by the time series in the defining set are experiencing large deviations from their respective trends. Following the reasoning above we expect these deviations to be caused by some physical phenomenon. Although each physical variable will not be perturbed at exactly the same time or for the same duration, we expect some time overlap of deviations in affected fields. The gamma method identifies such times (see the Figures in section 2.4). Time periods near these maxima are defined as features of interest for further study. Arranging the maxima in descending order produces a rank-ordered set of time extents as identified features $\{\mathcal{F}_1, \mathcal{F}_2, \ldots\}$, where the ranking is essentially by size of overlap. See the accompanying tutorial codes for a constructed example.

By construction this set of features is dependent on the choice of defining set, which allows tuning of the method for specific phenomena. The analyst uses their knowledge of the physical context to decide which time series to include in the defining set, an appropriate trend, and how to synchronize the time series to one another. The chosen time series must then be scaled so that they may be compared in $\gamma(t)$. Finally, the feature length must be chosen. We consider each step in turn.

2.3.2 The Defining Set

The defining set can be chosen any way the analyst sees fit. If the analyst is looking for a specific physical phenomenon, only the fields whose deviations would be associated with those events are included in the defining set. Alternatively the method may be applied to various subsets of the available time series to identify features first, with the analyst supplying physical explanations afterward.

The analyst may construct any time series they deem useful and include it in the defining set. For example, suppose two thermistor chains are deployed in a small lake. The thermistor chains each produce a vertical vector of temperature time series. If all temperature time series are included in the defining set the corresponding $\gamma(t)$ has maxima when there is a temperature deviation at all sensors simultaneously. This choice of defining set may identify periods of temperature change driven at the lake scale, such as a deviation of temperature due to seasonal change. If instead the phenomenon of interest is a cold water inflow, it may suffice to take the depth-averaged values at each

chain and consider the difference of the two averaged time series as an indicator. Any time series the analyst can think of, and whose deviation would serve as an indicator for the given physical context and problem, may be included in the defining set. This would include smoothed versions of existing time series which preserve the relevant deviations [45], as well as time series produced from standard methods like EOF (i.e. amplitude time series) and scale-averaged wavelets if the analyst deems it appropriate [64].

Once the defining set is chosen, a trend must be chosen for each time series. If the trend is unknown, mathematical methods such as [67] may be used to identify it, but this is not always necessary. The time mean $\mu_i(t) = \langle x_i(t) \rangle_t$ is a reasonable constant valued choice in many applications. This is the choice we make for both data sets in section 2.4.

Finally, the defining set must be synchronized. Different sensors may have different sampling rates, deployment duration, etc. The analyst uses their knowledge of the instruments and physical context to arrange the time series from each sensor along some global time regime. This global time regime t is the time on which $\gamma = \gamma(t)$ depends. Differences in sampling rate may be handled by interpolation or subsampling, differences in duration by truncation to an appropriate overlapping time period, and so on. Once the defining set has been chosen and synchronized, the scaling must be chosen.

2.3.3 Scaling

Equation 2.1 includes a scaling constant for each absolute deviation series for two reasons. First, equation 2.2 defines $\gamma(t)$ as the minimum of all absolute deviation time series at every point in time. For this to make any physical sense every time series in the defining set should be nondimensionalized because each of them are sampled from physical quantities having possibly different units. Second, the choice of nondimensionalization constant κ_i allows further tuning of the method. Scalings may be chosen to increase the influence of some physical quantities on $\gamma(t)$ while decreasing the influence of others. Care must be taken here, because scaling a curve by a larger value of κ increases the maxima of the corresponding curve, and decreases its effect on $\gamma(t)$. Therefore curves whose perturbations are considered more important in the given context should be scaled down, not up. For the examples given in section 2.4 we have chosen to scale each time series by their respective maximum values, which corresponds to the assumption that all perturbations are equally important. In general, the choice of scaling is another opportunity for the analyst to apply their knowledge of the context and tune the gamma method to their purposes.

2.3.4 Feature Length

Once the analyst has chosen the defining set, trend, synchronization, and scalings, the final choice is feature length l. This parameter is simply an approximate length of time that the physical phenomena of interest is expected to last. In our algorithm, we use a windowing procedure, where maxima of gamma are identified, and features are defined as the time window of length l whose midpoint is at the maxima. If the feature length is unknown, then l may be set to be very short so that features identify maxima in gamma. This is a parameter that can easily be tuned after the $\gamma(t)$ curve is found.

2.3.5 Feature Identification

The work in previous sections allows us to write problem 2.3.1 as:

Mathematical Problem Statement. *Given a defining set consisting of time series* $\{x_i(t)\}_{i=1}^k$ *synchronized along a global time regime, with respective scaling constants* κ_i *and trends* $\mu_i(t)$*, form*

$$\gamma(t) = \min_i \{\kappa_i |x_i(t) - \mu_i(t)|\}.$$

Identify rank-ordered features $\{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_r\}$ *as time windows of length* l *centred at the local maxima of* $\gamma(t)$*.*

We solve this problem iteratively, allowing overlapping features. Note that this means, for example, that the top several maxima of $\gamma(t)$ may all be included in the first feature. In that case the second feature would not be centred at the second highest global maximum, but rather at the highest maximum outside the first feature.

Problem 2.3.5 is solved using Algorithm 1. The rank-ordered identified features $\{\mathcal{F}_1, \mathcal{F}_2, \ldots\}$ are generated by iteration on the the maxima of $\gamma(t)$. MATLAB codes implementing Algorithm 1 were used for all results presented in section 2.4. Tutorial codes in MATLAB, R, and python are included in the supplementary material of the publication.

2.4 Results

2.4.1 Monterey Bay

The first data set we will consider is from a moored array of instruments deployed in the nearshore coastal environment of Monterey Bay, Cailfornia from July 7–21, 2011. The

Algorithm 1 Identify Features

load, clean, and filter data
choose defining set with trends, synchronization, and scaling
choose feature length l, and number of features r
define $\gamma(t)$
for i = 1 to r **do**
  find $\gamma(t)$ maximum $\gamma(t_i)$
  set $\mathcal{F}_i$ to be the time extent of length l centered at t_i
  set $\gamma(\mathcal{F}_i) = 0$ so a new feature is found in next iteration
end for
return $\{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_r\}$

moored array measured density (derived from temperature and conductivity measurements) and velocities throughout the water column. For a detailed analysis of this data set see [65]. High-resolution measurements were collected near a persistent upwelling front that forms between recently upwelled waters and warmer stratified waters that are trapped inside the bay (termed an upwelling shadow front, Fig 1a of [65]). The front propagates as a buoyant plume front past the instrument array with high kinetic energy before breaking up into a combination of large amplitude internal waves and instabilities.

Both density ρ and kinetic energy $KE = \frac{1}{2}(u^2 + v^2 + w^2)$ (omitting ρ_0) are useful for identifying fronts, internal waves, and instabilities. The overlap of the time series of both quantities has dimensions $M \times N = 35 \times 19701$ where M is the number of points in depth z, binned 0.5 m apart, and N is the number of samples in time t, taken every minute. Each of the vector-valued time series for ρ and KE are comprised of 35 time series, for a total of 70 individual time series. The gamma method may be applied directly to these 70 series, but a much simpler choice is appropriate in this context. The large kinetic energy and density events of interest tend to induce changes in the whole portion of the water column sampled by the data set. This makes the depth averaged means $\overline{\rho}$ and $\overline{KE}$ good indicators. These are 2 time series of length N, and we take them as our defining set. These time series are already synchronized because we expect fronts, internal waves, and instabilities to cause deviations in ρ and KE nearly simultaneously. We also scale each of the deviation series by their maximum values since we consider both to be equally important. These choices then define $\gamma(t)$. Based on known forcing associated with local diurnal winds (cf. [65]), we define our feature length as a day.

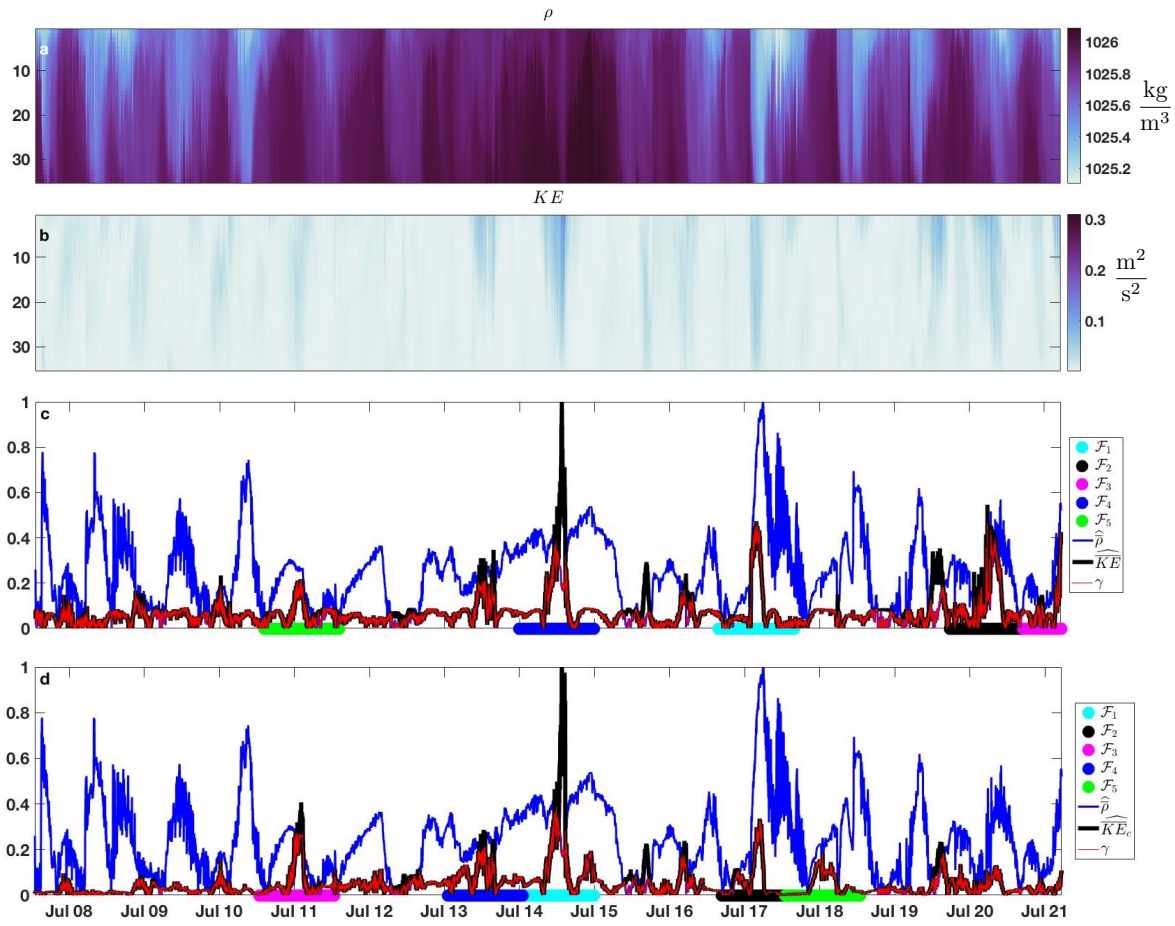

Figure 2.2: The gamma method applied to the Monterey Bay data set. Panel a shows the full density ρ (kg/m^3) and panel b shows the full kinetic energy KE (m^2/s^2). In both a and b the vertical axis is bin number. Panel c shows the results of the gamma method using the defining set $\{\overline{\rho}, \overline{KE}\}$, and panel d shows the results of using the gamma method using the defining set $\{\overline{\rho}, \overline{KE}_c\}$. All panels are aligned along the global time regime indicated below panel d.

Figure 2.2 panel c shows the result of applying the gamma method. Panel c shows the first five features $\mathcal{F}_i$. Notice the most important feature, $\mathcal{F}_1$, corresponds to the frontal crossing of July 17, a feature identified and studied extensively in [65]. In [65], this particular event was identified based on a more complicated filtering and wavelet analysis of the data set. Features $\mathcal{F}_2$ and $\mathcal{F}_3$ are large frontal crossing and internal wave events, and $\mathcal{F}_4$ coincides with a large regional-scale upwelling event and delineates a difference in

forcing relative to earlier events (see discussion in [65]). The next most important feature is $\mathcal{F}_5$. The density profile, along with the velocity data (not shown) indicates that this feature is an across shore pulse of cold water (see [65] Figure 1 b for orientation of axes). This is an example of a feature which may not have been identified by an analysis that did not use the method.

Figure 2.2 panel d shows the result of applying the gamma method using $\overline{\rho}$, and an alternate choice of a second time series. Stratification stabilizes the water column. When kinetic energy is high but stratification is weak, we expect more vertical mixing. To capture this idea, we define the conditioned depth averaged kinetic energy, $\overline{KE}_c$ as

$$\overline{KE}_c = \frac{\overline{KE}}{|\rho_B - \rho_T|} \tag{2.3}$$

where ρ_B is the density at the bottom sensor, and ρ_T is the density at the top sensor. $\overline{KE}_c$ is larger when the stratification is weak. The defining set is $\{\overline{\rho}, \overline{KE}_c\}$, so that the method is identifying times of density change with vertical mixing. Applying normalization by the maximum as before defines $\gamma(t)$, leading to the results shown in Figure 2.2 panel d. Note that $\mathcal{F}_1$ is now the upwelling period from July 14th to 15th. The large frontal crossing on July 17 is still identified as $\mathcal{F}_2$. This shows that important features may persist under time series conditioning. The across shore pulse of cold water is now identified as $\mathcal{F}_3$, because stratification is weak during this period. $\mathcal{F}_4$ is also a newly identified feature that is likely driven by strong surface wind forcing, due its confinement to the near-surface region. Finally, $\mathcal{F}_5$ identifies a time when $\overline{KE}$ is small, but the stratification is weak and the water is cold: this is another weakly stratified cold water pulse. Both cold water events $\mathcal{F}_3$ and $\mathcal{F}_5$ are not immediately clear from panels a or b of Figure 2.2, because the eye is drawn to other events. In this way the gamma method identifies features previously identified by analysts, but may also identify features that analysts miss.

2.4.2 Yax Chen

For the second example, we apply the gamma method to a data set from the submerged Yax Chen Cave System, in Tulum, Quintana Roo, Mexico. The Yax Chen Cave System is part of the larger Ox Bel Ha Cave System. The data set consists of time series from pressure (p), conductivity (s), and temperature (T) sensors deployed within Yax Chen from May 2016 to April 2018. The sensors were deployed as a follow up to the work presented in [7] in order to observe the changes in the aquifer as a result of heavy rainfall

events from hurricanes and tropical storms, which are common to the region. The sensors were deployed 10 m downstream from a cenote (a sinkhole connecting the surface to the submerged cave system) at a depth of 4 m. There was a single sensor for each physical quantity, and the three sensors sampled simultaneously every 30 minutes, so the time series are synchronized. Each time series has dimensions $M \times N = 1 \times 33697$ so there is no need to reduce the spatial dimension in this case. Normalization is taken by the respective maxima, and the feature length as one week.

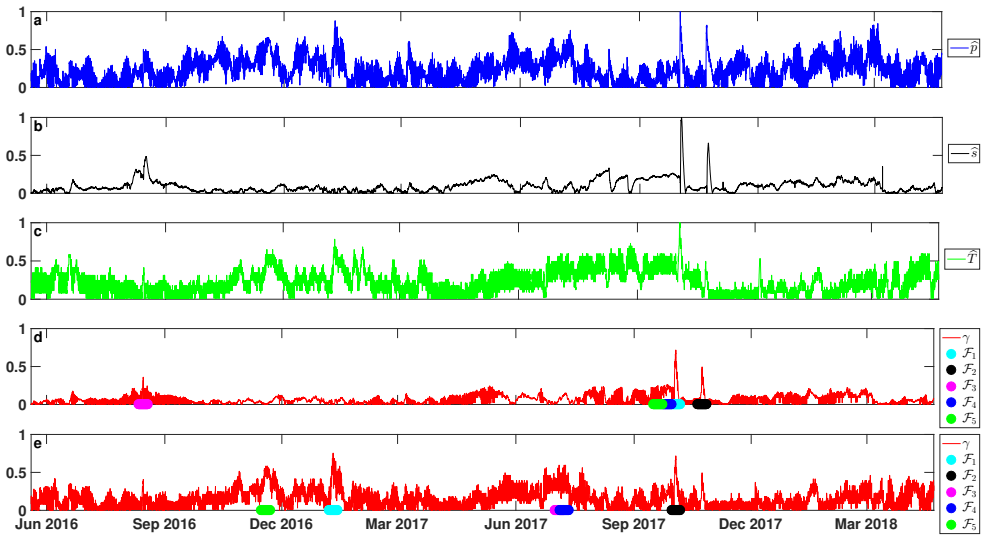

Figure 2.3: The gamma method applied to the Yax Chen data set. Panel a shows $\hat{p}$, panel b shows $\hat{s}$, and panel c shows $\hat{T}$. Panel d is $\gamma(t)$ for the defining set $\{p, s, T\}$. Panel e is $\gamma(t)$ for the defining set $\{p, T\}$.

Figure 2.3 panel d summarizes the results of applying the gamma method using the defining set of $\{p, s, T\}$. The early October 2017 event, corresponding to hurricane Nate[1] is identified as $\mathcal{F}_1$. The late October event, corresponding to hurricane Philippe is identified as $\mathcal{F}_2$. The mid August event corresponds to hurricane Earl, identified as $\mathcal{F}_3$. The last two features $\mathcal{F}_4$ and $\mathcal{F}_5$ identify the time period from mid to late September in which several storms, including hurricanes Irma and Jose could still have been affecting changes in the parameters measured in Yax Chen. This choice of the defining set identifies rainfall events large enough to affect pressure, salinity, and temperature in the cenote.

[1] All hurricane dates retrieved from the National Hurricane Center (https://www.nhc.noaa.gov/)

Figure 2.3 panel e summarizes the results of applying the gamma method using the defining set of $\{p, T\}$, i.e. without salinity. Since variations in salinity can only be due to mixing with the underlying marine water, this choice of defining set allows for the identification of events associated with longer trends, as opposed to turbulent mixing events [7]. Features $\mathcal{F}_1$ (early January 2017) and $\mathcal{F}_5$ (mid November 2016) correspond to large rain events that are not hurricane related. Early October 2017, $\mathcal{F}_2$, corresponds to hurricane Nate. A hurricane's primary expression in the cave network is via the turbulent mixing between the meteoric lens and the underlying marine water mass, resulting in variations in s, but s is not included in the defining set. This explains why hurricane Nate is not identified as $\mathcal{F}_1$, and also why Hurricane Phillippe is not captured. Features $\mathcal{F}_3$ and $\mathcal{F}_4$ (first half of July 2017) do not coincide with large rainfall events, and their identification by the gamma method as features which merit further study is completely new.

2.5 Discussion & Conclusions

Section 2.4.1 shows that the gamma method is able to automatically identify features of interest previously identified in an ad hoc manner, while also identifying new significant events. This means the gamma method can be applied to previously studied data sets and may find new results. Section 2.4.2 shows that the gamma method may be applied as soon as the physical context is known, to identify a set of features worthy of further study. Both examples outline how the analyst uses their knowledge of the physical context to choose the defining set, trend, scalings, synchronization, and feature length. For the sake of presentation we have outlined a broad range of possible necessary steps for choosing and synchronizing the defining set, but the practical application of the gamma method to a particular data set needs only a few steps. In practice we have found that taking the trend set to be the time mean and scaling by respective maxima serve as good default choices.

The gamma method depends on the overlap of perturbed fields. For short-duration features, or time series from sensors spaced far apart, it may be beneficial to time lag the time series before applying this method. For example, using the example of two thermistor chains in a lake from section 2.3.2, if the analyst is interested in temperature changes due to inflow, water masses inducing the change in temperature may pass the two thermistor chains separated by some time lag. In this case it may be preferable to make the defining set to be all of the sensors, but with an appropriate time lag on time series from one of the chains. If time lags are unknown but suspected, it may be possible

to infer them by brute force application of the gamma method to a range of possible time lags. Finding the time lag appropriate for a given time series is a highly field- and application-dependent problem and so must be left to the analyst, or other methods.

If the knowledge of the physical context is incomplete, so that expected phenomena or time lags for synchronization are unknown, a modified version of the method may still be applied as follows. The defining set should include many, if not all, of the available time series. Since the phenomena and time lags are unknown, it may be that a feature of interest perturbs some but not all time series at a given time. The gamma method presented above is inappropriate, because a single time series being unperturbed will cause the method to miss the feature altogether. There is a simple fix for this: define $\gamma(t)$ not as the pointwise minimum of the deviation series (equation 2.2), but as some suitable intermediary curve. For example if the method is applied to a defining set with 10 time series, it is probably worth investigating features which result from the deviation of 8 of them, so $\gamma(t)$ could be taken as the third from minimum curve. Taking an intermediary curve for $\gamma(t)$ also ameliorates the problem of faulty or intermittent sensors. Note this modification essentially ignores time series whose time lags cause them to be unsynchronized with the rest of the data set. The level of the intermediary curve is another parameter that may be swept. In general, the weaker the knowledge of the analyst, the more parameters there are to sweep. This would also apply, for example, to the scaling constants. The code runs on the order of seconds on modest hardware on all data sets we have tried, and is easy to parallelize for larger data sets or large sweeps, as necessary.

There are many other immediate possibile extensions of the gamma method. If positive and negative deviations from the mean are not equally important, the definition of gamma may be changed to a signed deviation instead. If the data is streaming rather than complete, the method could be applied with a trend μ defined by an appropriate recent window, resulting in an analogue of more sophisticated methods such as those presented in [19]. Features could be chosen by looking for extended deviations of $\gamma(t)$, rather than maxima. The reader may have noticed any number of immediate modifications that could be made to the method as it was presented.

Hurricane Nate's identification over both choices of defining set in section 2.4.2 suggests that the gamma method could be employed to identify important features by their persistence across choices of defining set. Persistence over a parameter sweep is used as a measure of a topological feature's importance in topological data analysis (see section 2.4 of [6] for an intuitive explanation). The gamma method could be run multiple times to sweep the choice of defining set as the parameter, yielding a final output of the most frequent features across all choices of the defining set. These persistent features would

then be candidates for closer study.

Clearly the gamma method is not as mathematically sophisticated as some other options. It is not designed to outline spectral information, identify weak synchronous signals, or automatically identify correct time shifts or choose the correct scaling. More sophisticated methods such as [37] address all of these concerns. Nevertheless, even those readers with the resources to confidently apply one of the many vector time series methods available to yield results they are satisfied with may find the gamma method useful as a diagnostic. In many cases we have found that the gamma method's incredible clarity and speed make it worth running before more sophisticated methods. For example the gamma method may be used to define time periods in a data set on which other methods are applied. Continuing the lake example, the method could be used to identify features defining cold and warm time periods before applying conventional methods to the data within those time periods. The results of the conventional methods may then be compared and contrasted across different time periods. The advantage of this process is that the time periods are defined mathematically, rather than by visual inspection.

In summary, the implementation of the gamma method to a given data set is straightforward and computationally inexpensive. The method is flexible and transparent, which allows it to be employed in a wide variety of contexts, and easily modified as necessary. After the initial tuning of the choices for a given context and problem, the method automates identification of a set of features which are worthy of further study.

Chapter 3

Features in Time-Indexed Model Output

Our initial problem was to identify features in time series data sets: data consisting of multiple time series each sampling a different physical variable at a single location. This is a common type of data set generated at an *in situ* instrument cluster. But what about other types of data sets? For example those consisting of many time series all sampling the same few physical variable at different locations. This type of data set is common in Computational Fluid Dynamics (CFD), where the output of a model run is a data set consisting of physical field values over many locations, at many times. CFD data is an example of time-indexed model output. Typically each physical variable in the model output is represented by many time series, one for each grid point, and far fewer time outputs. So while in the time series data set case we have a few long time series, in the time-indexed model output case we have many many shorter time series for each physical variable in the data set. More generally the fields in time-indexed model output simply consist of numerical values all having the same units. For the moment consider CFD data sets as a concrete example.

3.1 The Gamma Method Applied to CFD data

There is nothing keeping the gamma method from being applied to CFD data sets, but it must be done with care because the number of grid points in the simulation is often on the order of 10^8 or more. If we think of each grid point as the location of a virtual sensor

producing a time series of the given field at that location, that means the gamma method is being applied to a defining set chosen from thousands of time series. Put another way we are applying the gamma method to a time series data set with many more time series than we would expect from even a large deployment of field sensors.

One way to apply the gamma method to time-indexed model output is to construct a time series data set from it. This can be done using bulk measures of the domain to collapse spatial information in order to get a more manageable number of time series. Although they contain no spatial information, well chosen bulk measures can still identify features.

We consider a few examples. Both data sets were generated using a spectral collocation method called SPINS [56].

3.1.1 A Mode 2 Kelvin Wave

Figure 3.1: The rotation modified mode 2 wave discussed in [9]. The enstrophy field, which indicates energy dissipation, is shown. See text for details.

We first consider a data set from a simulation of a breaking Kelvin wave on the laboratory scale. To form these waves mixed fluid is initially separated from a stratified main portion of the tank by a barrier. The barrier is removed suddenly and various types

of internal waves are free to form. The stratification is chosen so that the dominant waves are mode-2 (some lines of constant density are displaced upwards while other are displace downwards). Rotation of the domain biases the wave ampitude to one side of the tank and leads to the generation of three-dimensional billows. The enstrophy field of the wave as it travels along the wall is shown Figure 3.1. For a full background see [9].

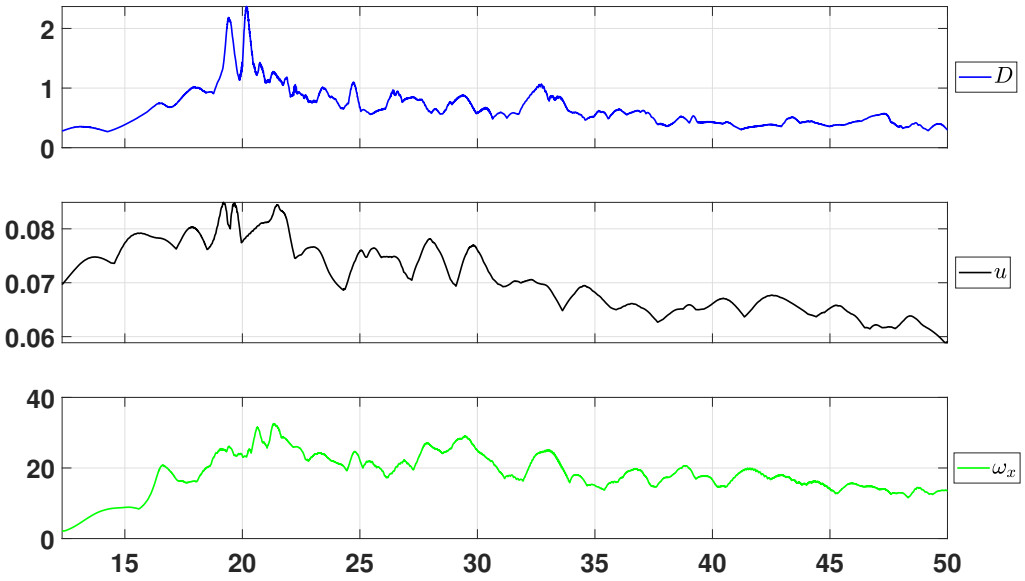

Figure 3.2: The data for maximum dissipation D, maximum horizontal velocity u, and maximum x component of vorticity ω_x. Note the different scalings for each quantity.

We extract the maximum viscous dissipation D as an indicator for activity at small scales, maximum horizontal velocity u as an indicator of large scale currents, and maximum x component of vorticity ω_x as an indicator of three dimensionalization to get a single time series for each physical quantity. Note in this case we were able to obtain higher time resolution in these time series by exporting values more often than the full field information, so that each of these time series was about 30000 outputs. These choices made, we have constructed a time series data set (Figure 3.2) with three dynamically relevant time series.

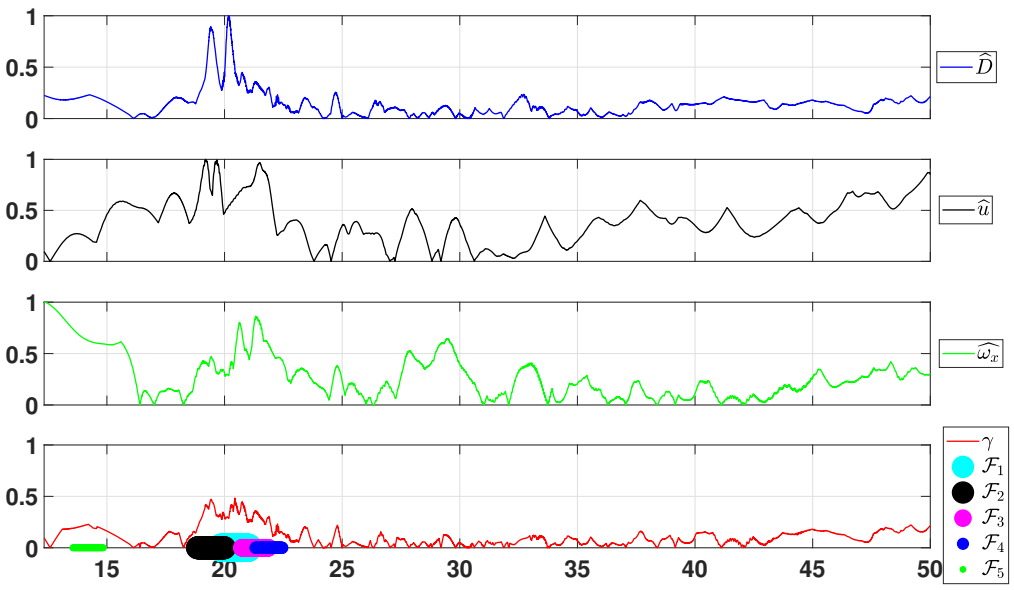

Figure 3.3: The absolute deviation series $\widehat{D}$, $\widehat{u}$, and $\widehat{\omega_x}$ with the gamma method results in the bottom panel.

Figure 3.3 shows the results of applying the gamma method to this data set, where all three time series are included in the defining set and the scaling is by the maxima of the respective series. The length of the feature was chosen to be 500 outputs, as an educated guess. The gamma method indicates an initial burst of instability just before that depicted in Figure 2 a of [9]. That is, gamma identifies the transition point around 20 s, as depicted in the middle panel of Figure 3.4. We also performed the gamma method with the maxima of the three components of velocity as the defining set, and another with the maxima of the three components of vorticity as the defining set, and derived similar results (not shown). Perhaps more time in [9] should have been devoted to the onset.

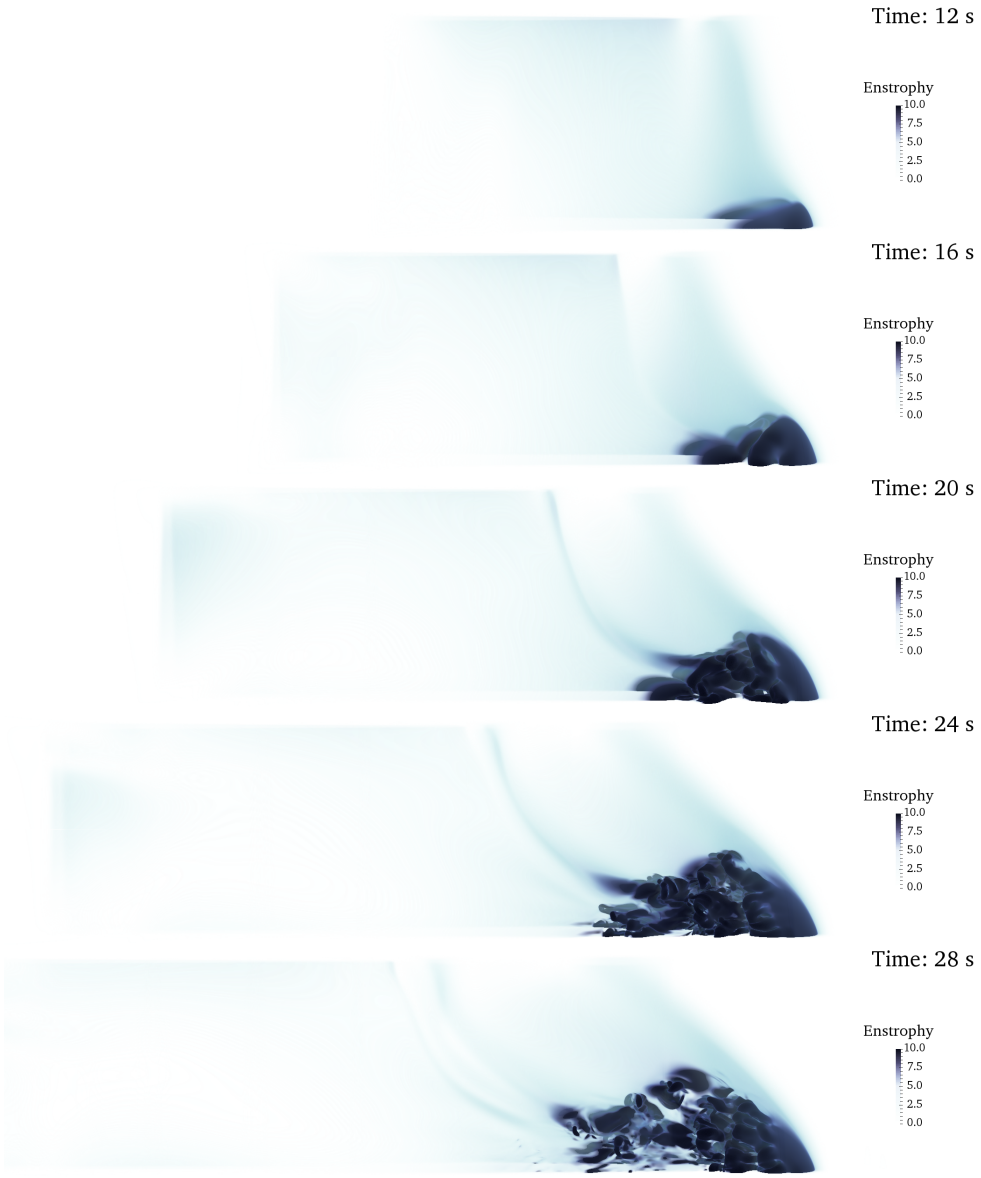

Figure 3.4: The evolution of the rotation modified mode 2 wave discussed in [9]. See text for details.

3.1.2 Internal Seiche with Multiple Instability Types

It is well known [31] that the diffusivity of salt is two magnitudes lower than the diffusivity of heat (so that heat diffuses much more quickly). Thus in systems stratified by variations in temperature and salinity (of which the ocean is a prominent example) resolving thin features involving salinity gradients is a significant challenge to the computational fluid dynamicist. While many idealized problems have been studied (salt fingers [53], thermohaline staircases [44]) the features were typically isolated from larger scale motions. We chose an internal seiche (standing wave) because it has a broad literature, a relatively simple laboratory implementation, and because it provides large scale currents that compete with any small scale instabilities that develop.

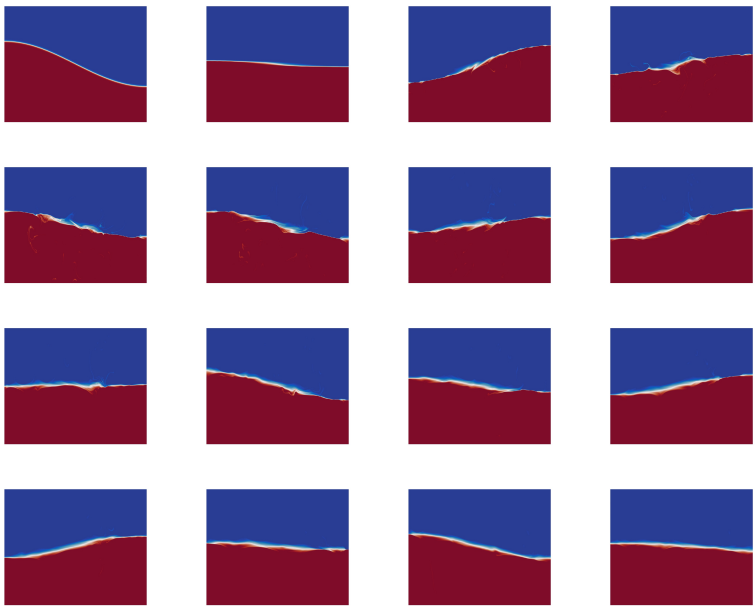

Figure 3.5: The evolution of the density field over the simulation every 15 outputs. Density values go from blue to white to red as they increase.

Figure 3.5 shows the "side-view" of the evolution of the internal seiche (with axes removed to keep the panels a reasonable size), while Figure 3.6 shows the detailed development of the instability at time 45. The density field is shown as it is the most

intuitive, but we are interested in the gamma method applied to velocity field diagnostics since these will have the clearest traces of the seiche (primarily horizontal velocities) and the instabilities (a mix of vertical and horizontal velocities).

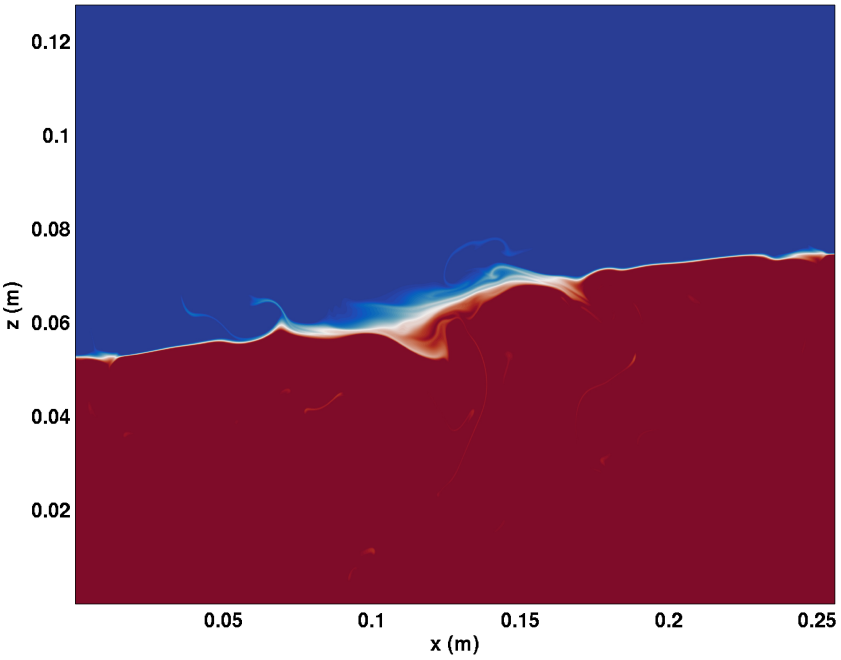

Figure 3.6: The top right panel of Figure 3.5, the density field at output 45, showing the development of the instability.

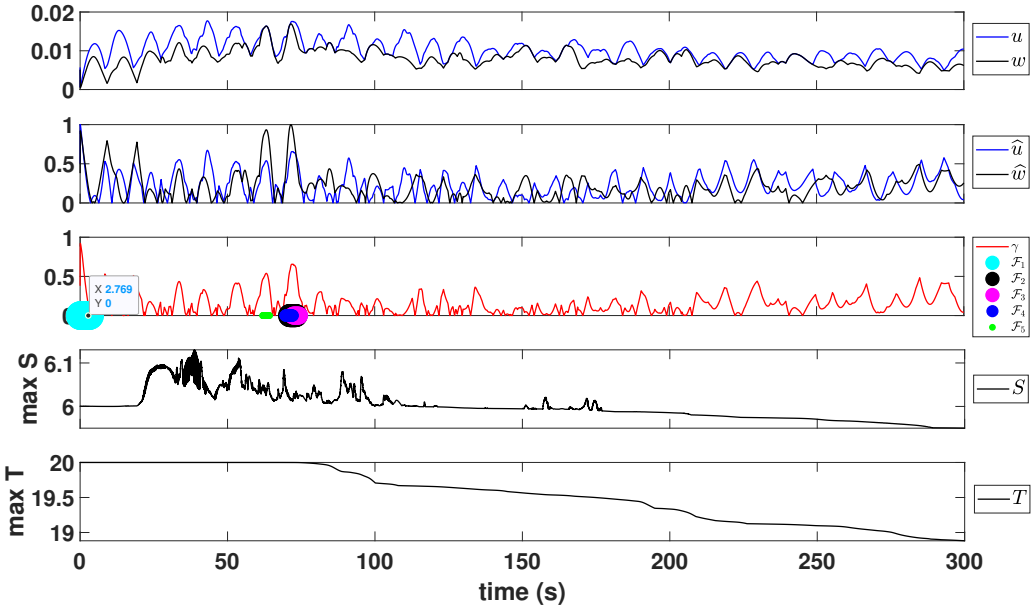

Figure 3.7: The gamma method applied to the seiche data set. See text for details.

Figure 3.7 shows the results of applying the gamma method to this data set, using the maximum horizontal velocity u and maximum vertical velocity w as the members of the defining set. The top panel shows the raw velocity data and the second panel shows the associated absolute deviation series resulting from scaling by the maxima of each series. Panel three shows the results of the gamma method. We note that $\mathcal{F}_1$ identifies the initial state, while $\mathcal{F}_i$ for $i = 2, 3, 4, 5$ are clustered around time 72. The gamma method has indicated a time when the maximum velocities are simultaneously maximized, as evident in the first panel of Figure 3.7. The reader may object, in this case, that we might have used a kinetic energy series formed from the maxima series instead. However that doesn't really make sense as the maxima of the velocities need not occur at the same point, and so forming the sum of their squares is not likely the kinetic energy at any grid point in the simulation.

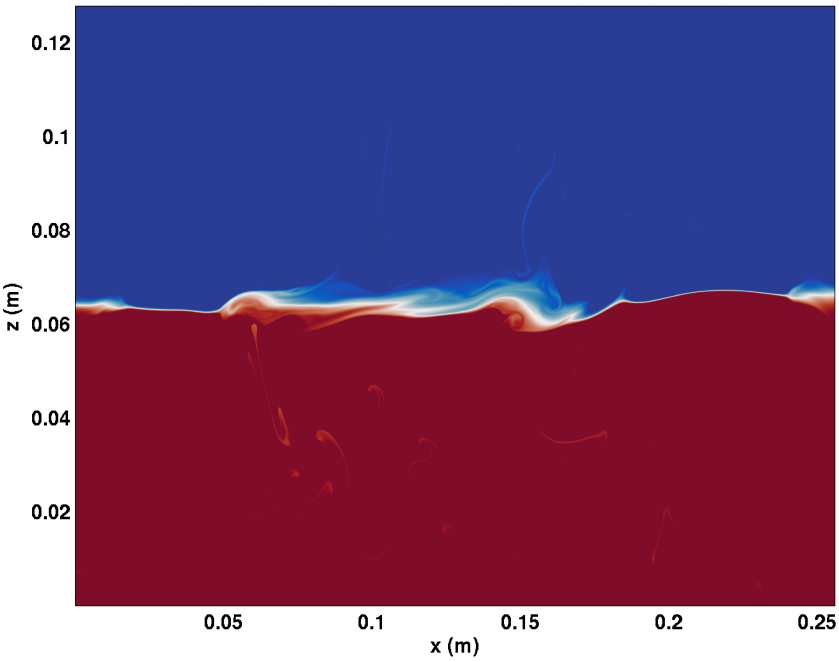

Figure 3.8: The density field at output 72, as chosen by the gamma method.

Gamma chooses time 72, as depicted in Figure 3.8. As a tutorial example, this again shows that the gamma method is identifying times of mathematical interest which may not be directly evident from visualizations. While the velocities were used in the analysis, flows are often visualized by their scalar fields, such as density or temperature. The horizontal pycnocline at time 72 could be mistaken as an indication of a queiscent fluid. Instead, in this case it is because the velocities are largest when all energy is kinetic rather than potential, as when a swinging pendulum passes its lowest point. This is not a time likely to be chosen by ad hoc means.

The bottom two panels of Figure 3.7 are included to further illustrate the usefulness of having a mathematical way to choose times of interest. It appears the maximum salinity S is somewhat under-resolved during the formation and development of the instability at the pycnocline, as we might expect given the numerical difficulties discussed above. However all series are well resolved during the features, and so we need not worry about the temporally under-resolved salinity (unless of course it is so extreme as to invalidate the rest of the simulation). This shows that the gamma method can assist in choosing

when to rerun experiments.

The maximum temperature T is included to reinforce the point that every series included in the defining set must be relevant to the phenomena of interest. As we are interested in times of maximized flow, temperature is not particularly relevant. Indeed in this case it is constant until after the features, and so its inclusion would have zeroed out the gamma curve, and no features would have been identified at all.

3.1.3 Summary

In summary, well chosen bulk measures allows the application of the gamma method, which once again serves as a diagnostic to identify features. For this reason we have implemented the gamma method parallel to our numerical experiments in SPINS. It is now standard practice for us to produce an associated time series data set consisting of dynamically relevant bulk measure time series at a high time resolution, alongside the relatively infrequent spatial time-indexed model output. The gamma method then runs automatically at the end of the experiment.

3.2 The Need for Another Method

The gamma method is designed to be applied to time series data sets including measurements from many physical fields. It is a way to cut through the clutter of a data set which contains multiple time series to isolate times of interest. In contrast time-indexed model output often consists of detailed spatial information of a few or even one physical field. In these cases the data set consists of primarily spatial information, and so to collapse this information using a bulk measure is to ignore almost all the information arduously generated using precious clock time. If there is only one physical field generated, a bulk measure time series data set consists of only one time series, and the gamma method is trivial. So while the gamma method can always be used on time-indexed model output by constructing a parallel time series data set, much of the information present is ignored. It turns out that there is no straightforward way to apply the gamma method to time-indexed model output in a way that takes into account the spatial information present. To see why, consider the following example of a CFD data set.

Figure 3.9 shows a few time outputs of 2D CFD data simulating an internal wave train in a spatially varying wave guide. This is generated by the numerical equivalent of what

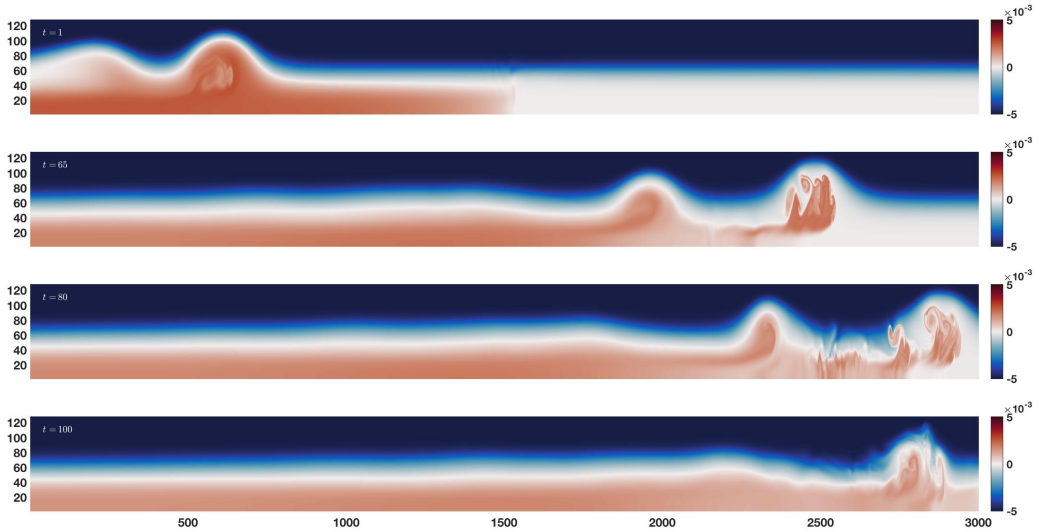

Figure 3.9: An internal wave train propagates from left to right and encounters a sharp change in the background density profile. The density field is shown, so that the density change around 1500 is clear.

experimentalists refer to as a lock release: fluid of a set density is suddenly released from behind a barrier and is allowed to freely form waves in the stratified tank. The particular situation is set up so that a wave train of internal solitary waves with a trapped core forms, propagates some distance and then encounters a sharp change in the background density (a pycnocline). This change removes the near bottom stratification, while the main wave guide remains unchanged. To the best of our knowledge, there is no *a priori* theory for the wave evolution in this cases and we find that the change in the near bottom wave guide leads to the destruction of the trapped core in the leading wave. This, in turn, leads to a significant increase in short length scale activity and a loss of material from the leading wave, and a significant perturbation to the second wave in the wave train. In this case there is no readily apparent way to define a "base" flow, since even prior to the collapse of the core, the disappearance of the near boundary wave guide implies a core cannot persist [34]. Throughout the thesis this case will be referred to as the "dual pycnocline" case.

The gamma method is designed to find interesting times. In Figure 3.9 it seems clear that the breakdown of the lead wave at time $t = 65$ or perhaps the vortex shedding from

the lead wave to the rear wave at $t = 80$ would be interesting here. The gamma method is not well suited to finding these times as it is not designed to work with this level of spatial information for a single field. The gamma method allows an analyst to use their domain knowledge, and the less knowledge they have, the more parameters must be swept. If little is known of the data set or its context, all choices of the defining set need to be considered. This is practical for time series data sets but becomes impractical for time-indexed model output because there are simply too many grid point time series in the model output. In this case there are over 360 000 grid points in the simulation, and a time series for each one, all sampling density fluctuations. As is clear in Figure 3.9 some grid point time series, for instance along the top of the domain, would have a near constant value throughout the simulation. The inclusion of even one constant time series would lead to a zero gamma curve and the method would fail. Moreover even if all constant grid points were avoided, the fluctuations of the time series at different grid points would occur at different times because the wave train is moving. We could again avoid this by taking a vertical strip of grid points somewhere in the domain, for example, but then all the gamma method would show was when the wave train would pass that strip. There is no *a priori* way to choose which grid points to include as time series in the defining set in order to identify the small scale structures of the breakdown or shedding. We did push the gamma method reasoning to its limit, applying it to this data set to find interesting locations within each time output, to some moderate success as a visualization method in stratified flow dynamics, but this extension is outside of this thesis' main themes. The interested reader can refer to Appendix section A.1 for these results. Clearly another method for finding interesting times in time-indexed model output would be useful. As with the gamma method we will proceed from first principles to construct one.

Chapter 4

The EOF Error Map Method

4.1 Author's Note

This chapter (4) originally appeared as [51], but the presentation here is greatly expanded with the addition of sections 4.3.1, 4.3.2, 4.3.4, and 4.3.5. A small amount of this content is re-organized material from [51], but most is completely new. As in section 2, there are other small differences throughout in order to help match the prose to the rest of the thesis, and provide some additional explanation where required.

4.2 Introduction

We present a data-centric diagnostic for identifying time subsets of model output which are worthy of further study. To minimize the cost of uptake and maximize the clarity of the presentation we have built this diagnostic on Empirical Orthogonal Functions (EOFs), which are used in an enormous variety of contexts (e.g. [28], [26], [4], [29], [17], etc.) and have implementations in every commonly used software toolbox (e.g. Matlab, R, Scipy). The method presented here can be applied to any data set for which an EOF analysis would be appropriate, but we will focus on the application to CFD data sets. The method is data driven, using a novel construction: a map of the EOF reconstruction errors as a function of time and the number of modes in the reconstruction. The interpretation of this EOF error map yields the identification of interesting times in each field in the data set for the cost of one Singular Value Decomposition (SVD) and one norm calculation per time output and choice of reconstruction.

The mathematical ideas behind EOFs have a long history, originating with [42], and go by many names, including Principal Component Analysis (PCA), Singular Value Decomposition (SVD), and Principal Orthogonal Decomposition (POD), depending on the community. These methods produce an orthogonal basis for the state space of a data set, where the basis vectors (EOFs) are rank-ordered by the amount of variance of the data they capture, as recorded in the eigenvalue for each basis vector. In particular, when the data has units of velocity, the variance has units of energy, so the basis is rank ordered by energy captured. Following the common parlance, we will use "energy" and "variance" interchangeably. Since the use of all basis vectors fully reconstructs the data, and the basis is rank-ordered by energy content, this representation can then be truncated to provide a reduced order reconstruction of the data. This reconstruction captures the most energy contained in the original data set per basis vector added, on average [20]. Efficient reconstructions of data are often the goal in statistical analysis, where EOF methods are referred to as PCA. For a review from this perspective see [1].

EOF methods are common in the atmospheric science, oceanography, and climate science communities where there has been an attempt to relate individual EOFs either to physical processes or to normal modes of the system being sampled. Such efforts have had some success, for example in the study of the El Niño Southern Oscillation [72], North Atlantic Oscillation [23], and the Arctic Oscillation [59]. The focus on the first, or "leading", EOF can be viewed as the study of a an EOF reconstruction (heavily) truncated to include only the first mode. As mentioned, some large scale motions have been captured this way, and correspondences have been drawn between physical processes and the leading EOF. However EOFs form an orthogonal set, and thus adding subsequent EOFs to the reconstruction, while simultaneously expecting those additional modes to correspond to physical processes, is to assume that the physical processes or normal modes in question are orthogonal. This is not true in general. Instead, a kind of contamination occurs: [71] applied an EOF analysis to a constructed flow with multiple dominant structures. They found that EOFs roughly corresponding to specific fluid structures were contaminated by components of other structures (their Figures 3 and 6). Several modifications to EOF methods have been developed to produce modes which may have a more direct physical interpretation, but these methods often require a choice to be made, and it is not often clear which choice is correct. We refer the reader to the review by [17] of EOFs and their extensions for a history of these difficulties. In the error map method we simply use the standard EOF, as it is the most widely used. Moreover, we focus on the reconstruction perspective in order to build the EOF error map. This avoids the difficulties of focusing on individual EOFs outlined above. In addition, the

construction of the error map includes errors from every truncated reconstruction, so there is no need to consider the problem of choosing a particular EOF to focus on. Because it avoids focusing on either individual EOFs, or individual EOF reconstructions, the EOF error map method is different from every previous EOF-based method.

There are, of course, a wide variety of existing data analysis methods for CFD data sets which are not EOF-based, but none of them serve the same function as the EOF error map method presented here. There are local, Eulerian (i.e., measurements at fixed locations) methods to identify vortices based on the decomposition or invariants of the velocity-gradient tensor: the Q-, Δ-, and λ_2-criterions for example [30]. There are Lagrangian methods (i.e., based on moving particles) to identify coherent structures (e.g. transport barriers), such as those based on Finite Time Lyapunov Exponents [57], [58], or graph theoretic methods [15], [11], [48]. For a comparison of multiple Lagrangian methods applied to the same benchmark see [14]. There are a host of methods based on the spectral properties of the Koopman operator [40], and its finite dimensional approximation the Dynamic Mode Decomposition [49], which allow identification of structures in fluid flows based on the frequency of the structure's motion, such as the flapping frequency of a jet [50]. There are many reduced order methods besides EOF, including the related POD and Galerkin projection [46], [20]. For a review see [47]. In fact, there are many more analysis methods available which can be used to study CFD data sets. All of them make an *a priori* judgement on the field of interest (e.g. gradient of the velocity field, inter-particle separation, etc) and proceed with an analysis on that particular field in the data set. In contrast, the purpose of the EOF error map method is to identify interesting time periods within every field in the model output without an assumption on which variable is the most important. These features, in each field, then become targets for further study using any method appropriate, including those just mentioned.

Put another way, the EOF error map method is a diagnostic tool which is applied earlier in the analysis pipeline than the standard methods just discussed. As such it is not a competitor with those methods, but a way to facilitate their intelligent application. This is particularly relevant to large, coupled models in fields such environmental fluid mechanics involving biogeochemistry and climate modeling for which the CFD component is only a small portion of the model. Even sophisticated mathematical tools based exclusively on the fluid mechanics may miss an important event in one of the other components of the model (e.g. an algal bloom in the coupled model of a bay). Thus for large coupled models, we envision our method being applied as part of the model

execution, so that every field in the model output would be accompanied by identified features. Only the subsequent analysis would be discipline specific.

Error maps also carry a very low overhead. They are constructed directly from model output immediately after the completion of a numerical experiment and the only extra computational burden is the SVD and error map construction: there is no need to take derivatives of fields, it is not necessary to have particle data, there is no necessity to tune parameters in a graph theoretic clustering algorithm, etc. Error maps are used as a diagnostic to quickly identify features which should be investigated further, by whatever method is deemed useful for the particular application. This allows error maps to inform decisions on where higher overhead methods should be applied. In summary, the EOF error map is a low overhead method applied directly to model output as a way of focusing the application of other methods.

4.3 Empirical Orthogonal Functions

4.3.1 EOF From Discrete Data: Covariance Matrix Method

Let us proceed from first principles (see [33] Chapter 15). Suppose the data set has M grid points and N time outputs at times t_j, $j = 1, \ldots, N$. This is a sequence of snapshots $\{\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_N)\}$ where each $\mathbf{x}(t_j) \in \mathbb{R}^M$. Centre by the time mean, and make the resulting snapshots columns of a single matrix $\mathbf{X}$. Then the jth column of $\mathbf{X}$ is

$$\mathbf{X}_j = \mathbf{x}(t_j) - \langle \mathbf{x} \rangle \tag{4.1}$$

where the angle brackets indicates the time mean. The matrix $\mathbf{X}$ is

$$(\mathbf{X})_{ij} = x_i(t_j) - \langle x_i \rangle \tag{4.2}$$

where i indexes the grid points, j indexes the time outputs, and $\langle \mathbf{x} \rangle_i = \langle x_i \rangle$. Then $\mathbf{X}$ is an M by N matrix whose entries are time mean-centred time series of measurements at the grid points or sensor locations. We then form the covariance matrix

$$\mathbf{C}_\mathbf{X} = \frac{1}{N-1}\mathbf{X}\mathbf{X}^T$$

which is a symmetric M by M matrix with entries

$$\mathbf{C_X}_{ij} = \frac{1}{N-1}\sum_{k=1}^{N}(x_i(t_k) - \langle x_i \rangle)(x_j(t_k) - \langle x_j \rangle) \tag{4.3}$$

which is exactly the covariance of the point x_i with the point x_j. Along the diagonal, when $i = j$, this reduces to the variance of each x_i. It is reasonable to assume that those x_i with high variance values are sampling significant dynamic events. Similarly it is reasonable to assume that significant cross-covariance indicate a redundancy in the collected data (see section 15.3 of [33] for additional comments on these ideas). This indicates an opportunity for efficient lower-dimensional representation. Since $\mathbf{C_X}$ is symmetric and real, it is orthogonally diagonalizable with real, distinct eigenvectors. This means we can both maximize variance and minimize cross covariance by writing

$$\mathbf{C_X} = \frac{1}{N-1}\mathbf{U\Lambda U^T}$$

where $\mathbf{U}$ is the matrix whose columns are an orthonormal set of eigenvectors of $\mathbf{C_X}$. These columns are the M Empirical Orthogonal Functions ϕ_i which are also rank ordered with corresponding eigenvalues λ_i. Here $\mathbf{\Lambda}$ is diagonal and without loss of generality is rank ordered with the first eigenvalue being the largest: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M \geq 0$. To see why the eigenvalues are nonnegative, notice that for any column vector $\mathbf{z}$ we have

$$\mathbf{z}^T\mathbf{C_X z} = \frac{1}{N-1}\mathbf{z^T X X^T z} = \frac{1}{N-1}(\mathbf{X^T z})^\mathbf{T}\mathbf{X^T z} \geq 0$$

so that $\mathbf{C_X}$ is positive semidefinite. This is why the eigenvalues are nonnegative. We can now work in the EOF variable

$$\mathbf{Y} = \mathbf{U^T X}$$

where we find that

$$\mathbf{C_Y} = \frac{1}{N-1}\mathbf{Y}\mathbf{Y}^T = \frac{1}{N-1}\mathbf{\Lambda}$$

Physically, this means we have transformed to a basis formed by the EOFs where cross-covariances have been completely eliminated and the variances are rank ordered. So then λ_i gives the variance of $\mathbf{X}$ along the EOF ϕ_i.

Writing the data in the transformed variable we now have

$$\begin{aligned}(\mathbf{Y})_{ij} &= (\mathbf{U^T X})_{ij} \\ &= \phi_i \cdot (\mathbf{x}(t_j) - \langle \mathbf{x} \rangle)\end{aligned}$$

where we've written the result in terms of the snapshots $\mathbf{x}(t_j)$. Multiplying both sides on the left by $\mathbf{U}$ yields the data set as a projection onto the EOF basis

$$\begin{aligned}(\mathbf{X})_{ij} &= x_i(t_j) - \langle x_i \rangle \\ &= (\mathbf{UY})_{ij} \\ &= \sum_{k=1}^{M} Y_{kj} U_{ik} \\ &= \sum_{k=1}^{M} \left[\phi_k \cdot (\mathbf{x}(t_j) - \langle \mathbf{x} \rangle)\right] (\phi_k)_i\end{aligned}$$

where $(\phi)_i$ is the ith entry of the kth EOF. Letting
$[\phi_k \cdot (\mathbf{x}(t_j) - \langle \mathbf{x} \rangle)] = (\mathbf{U^T X})_{kj} = a_k(t_j)$ (notice $\langle a_k \rangle = 0$) we can then write the snapshots of the system as

$$\mathbf{x}(t_j) = \sum_{k=1}^{M} a_k(t_j)\phi_k + \langle \mathbf{x} \rangle \tag{4.4}$$

Since the vectors ϕ_i are orthogonal some call this the Proper Orthogonal Decomposition (POD). Written this way it is clear that the signal can be thought of as a mean signal with layers of corrections represented by the sum.

4.3.2 A Constructed Example

We now consider a few constructed examples for tutorial purposes. First consider a very simple case, where the data is

$$\mathbf{x}(t_j) = \begin{bmatrix} \sin(2\pi t_j) \\ \sin(2\pi t_j) \end{bmatrix} \tag{4.5}$$

which is $M = 2$ points for $t_j = 0, 0.001, 0.002, \ldots, 1$ which is $N = 1001$ outputs. See Figure 4.1 for a plot.

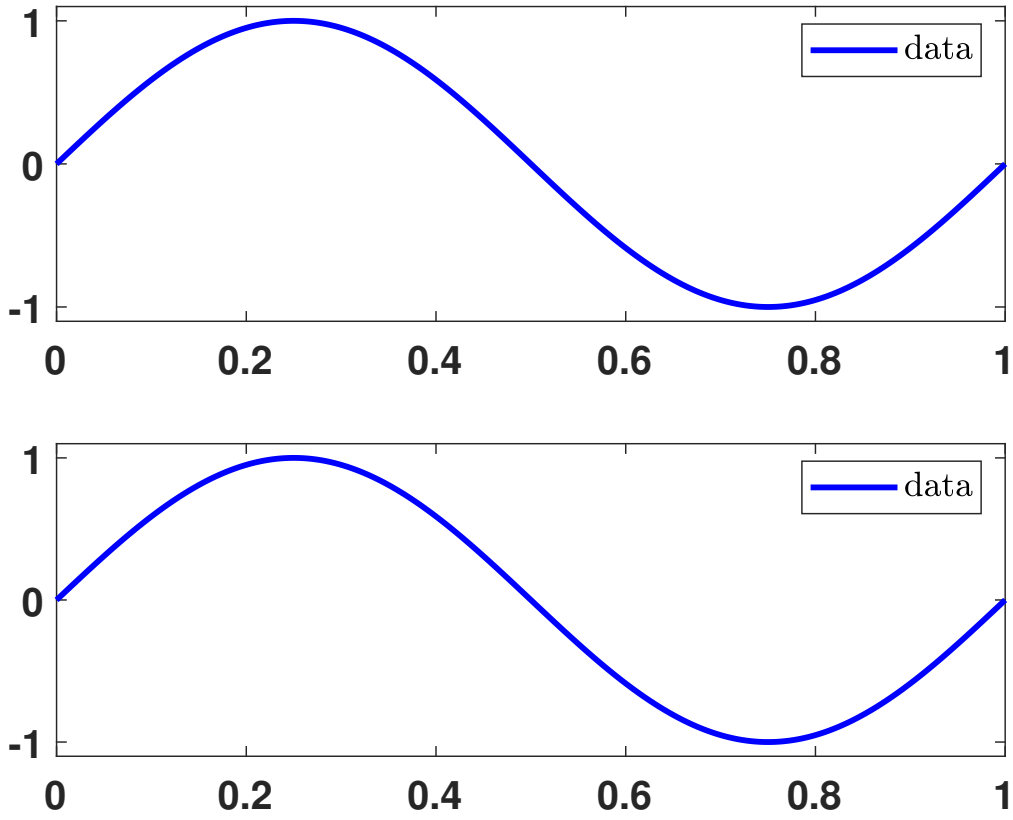

Figure 4.1: The data for the simple sine wave case of equation 4.5, with the first component in the top panel, and the second component in the bottom panel.

In this case $\langle \mathbf{x} \rangle = \mathbf{0}$ and we have from equation 4.14 that

$$\mathbf{x}(t_j) = a_1(t_j)\phi_1 + a_2(t_j)\phi_2$$

using the covariance method of section 4.3.1 to obtain the EOFs and coefficients shown in Figure 4.2.

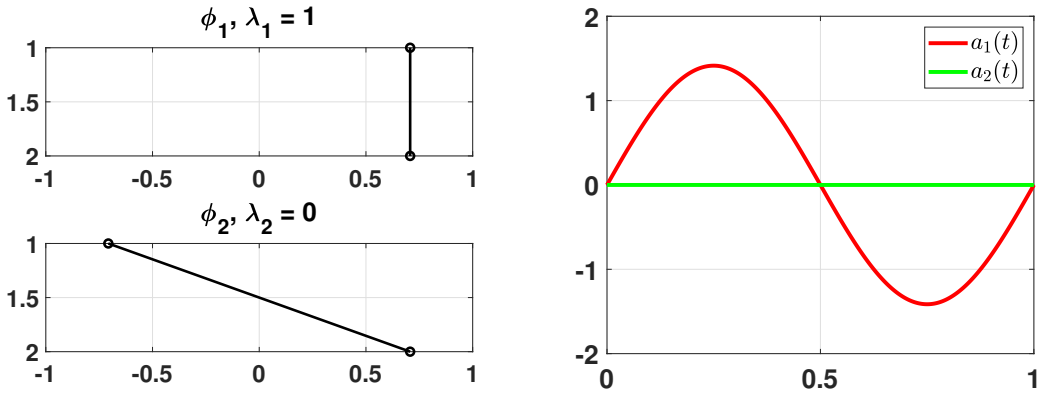

Figure 4.2: The EOF (left) and coefficients (right) for the constructed data.

Note that in this simple case, with both spatial points in perfect correlation, we have $\lambda_1 = 1$ and all of the energy is captured in ϕ_1. The shape of ϕ_1 is simply a constant value because the data is identical in both components. Since all of the energy is in ϕ_1, $\lambda_2 = 0$, and the ϕ_2 is irrelevant. This can be seen in the coefficients where $a_1(t)$ is a sine wave, and $a_2(t) = 0$. We see that the truncated reconstruction $a_1(t)\phi_1$ is sufficient in Figure 4.3.

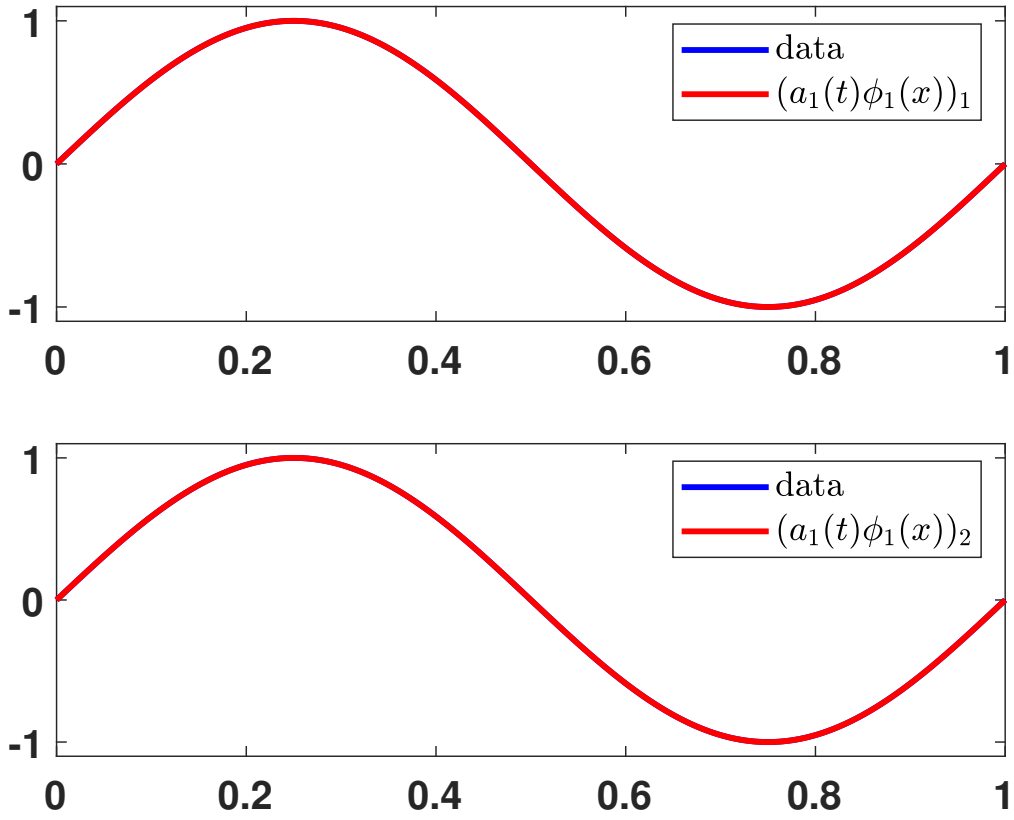

Figure 4.3: The 1 EOF reconstruction with the original data. The first component is in the top panel and the second component is in the bottom panel. Note that in this case because $\lambda_1 = 1$, the 1 EOF reconstruction is equal to the data (down to machine precision).

In the simple case of the perfect correlation of two spatial points $\lambda_1 = 1$ and the 1 EOF reconstruction is equal to the data down to machine precision. This makes sense because the time coefficient tracks the change over time, but the change over time is in perfect correlation. We now consider the case where the correlation is imperfect, taking the

simple data set just used and perturbing the second component by a Gaussian

$$\mathbf{x}(t_j) = \begin{bmatrix} \sin(2\pi t_j) \\ \sin(2\pi t_j) + \exp\left(-\left[\frac{t_j - 0.5}{0.01}\right]^2\right) \end{bmatrix} \tag{4.6}$$

which is $M = 2$ points for $t_j = 0, 0.001, 0.002, \ldots, 1$ which is $N = 1001$ outputs once again. See Figure 4.4 for a plot.

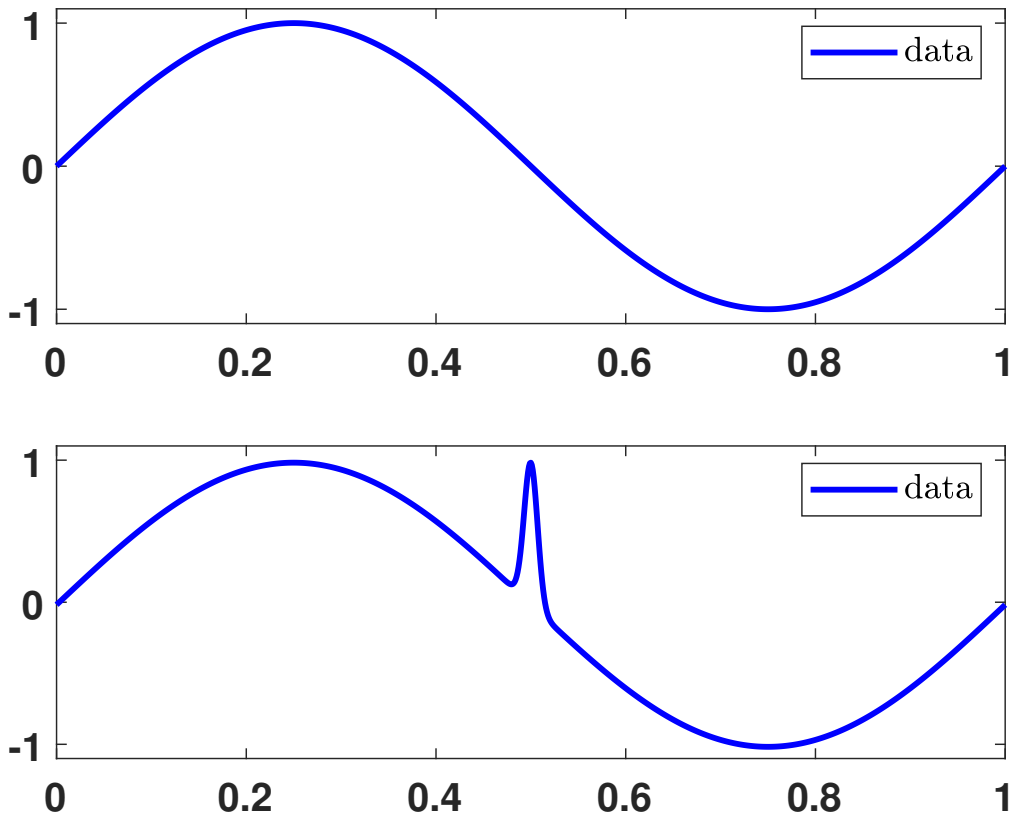

Figure 4.4: The data for equation 4.6, with the first component in the top panel, and the second component in the bottom panel.

Clearly the correlation is no longer perfect. Moreover due to the perturbation $\langle \mathbf{x} \rangle \neq \mathbf{0}$ in

the second component, and we have from equation 4.14 that

$$\mathbf{x}(t_j) = a_1(t_j)\phi_1 + a_2(t_j)\phi_2 + \langle \mathbf{x} \rangle$$

where now the EOFs and coefficients are those shown in Figure 4.5.

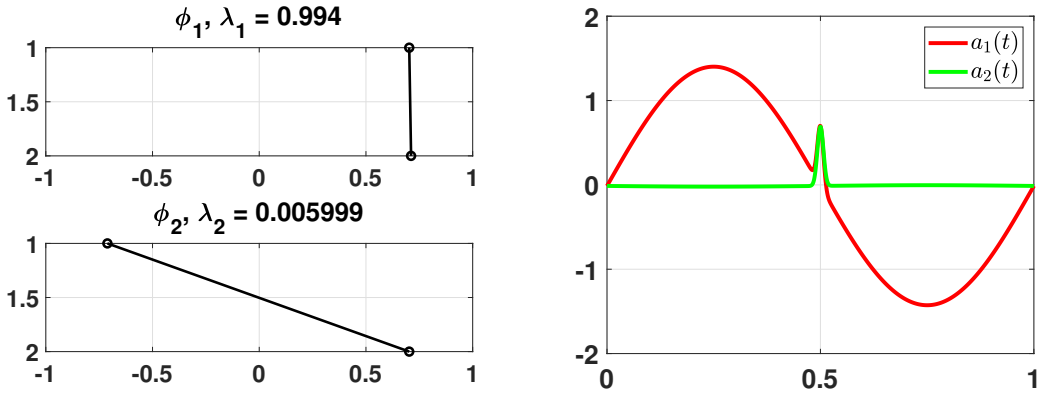

Figure 4.5: The EOF (left) and coefficients (right) for the constructed data.

We see that unlike the perfectly correlated case, there is now a small amount of energy in the second mode. The first coefficient $a_1(t)$ is no longer a sine wave but a sine wave perturbed. This reveals an important point about EOF reconstructions. Consider the following thought experiment. If we did not know what the closed form of the data was but had rather gathered it in some way, looking at the graphs of the two components, there is a clear separation of scales. If asked to write the data in two components we would probably write

$$\text{data } = (\text{long wave}) + (\text{local perturbation near } t = 0.5 \text{ in component 2})$$

If we were then asked to approximate the data by removing one addend from the sum we would almost certainly neglect the perturbation in the second component. Even without knowing the closed form of the data it is clear to us that the data is basically a sine wave in both components, with a perturbation in the second. We might hope, then, that the 1 mode EOF reconstruction would still be a sine wave, remaining unchanged from the reconstruction depicted in the simple case of Figure 4.3, and that the second mode would add in the perturbation. EOFs are not able to make such delineations. Instead the 1 mode EOF reconstruction attempts to capture the perturbation as well, as depicted in Figure 4.6.

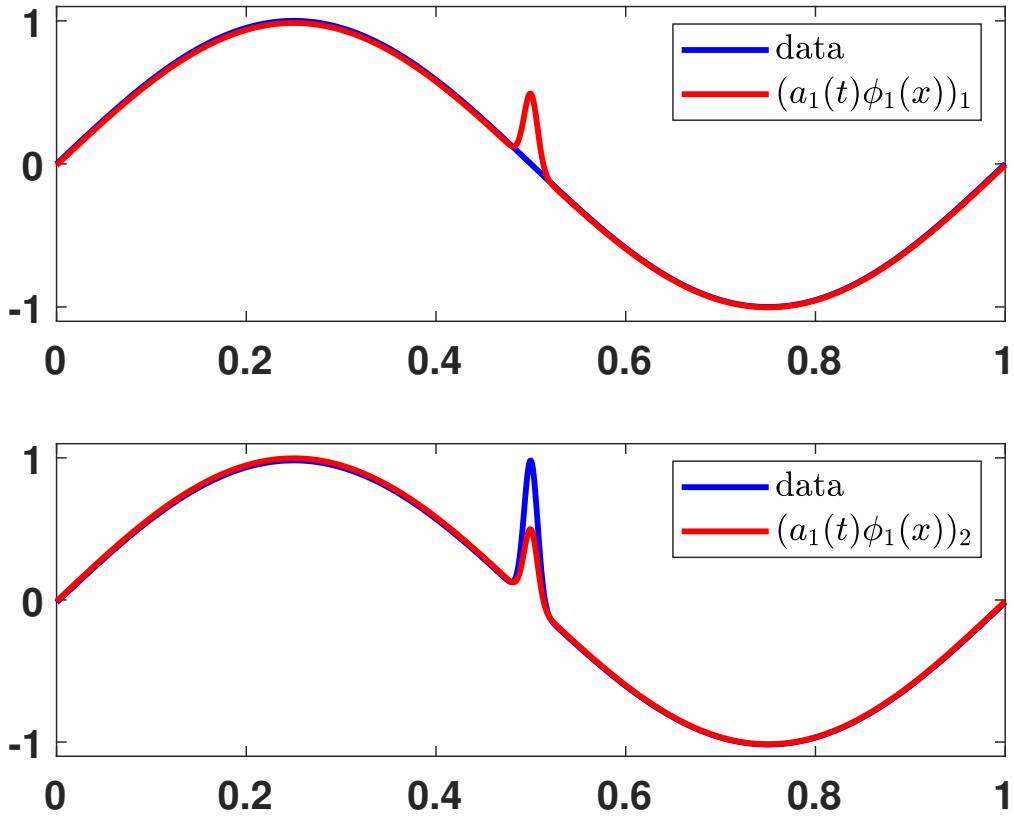

Figure 4.6: The 1 EOF reconstruction with the original data. The first component is in the top panel and the second component is in the bottom panel. Note that in this case the 2 EOF reconstruction would be equal to the data (up to machine precision).

EOF modes are selected by maximizing variance, and so do not necessarily result in a clean separation of scales. For the 1 EOF reconstruction to maximize variance the short scale perturbation cannot be ignored. In attempting to capture both scales in the first EOF the first component of the 1 mode reconstruction takes too large a value during the perturbation, and the second component takes too small a value during the perturbation. This explains the shape of ϕ_2 as depicted in Figure 4.5 because in the first component it must correct $a_1(t)\phi_1$ by subtraction, and in the second component it must correct $a_1(t)\phi_1$ by addition. This correction only occurs during the perturbation and so $a_2(t)$ is still zero

outside the perturbation.

We see that the perturbation's presence causes the first EOF to 'split the difference' between the components over the extent of the perturbation. So in fact the perturbation in the second component contaminates the first component if we use a 1 EOF approximation for the whole signal (see [71] for a discussion of contamination). This example illustrates why one should be careful of assigning physical meaning to individual EOFs. Moreover note that $a_1(t)$ and ϕ_1 have the wrong scale separately: $a_1(t)$ reaches a maximum of almost 1.5, and ϕ_1 reaches a maximum of around 0.7. It is their product which recovers the scale of the data. This is another reason we take the reconstruction perspective.

4.3.3 EOF From Discrete Data: SVD Method

We now present the SVD method of finding EOFs. Again beginning with a data set consisting of M grid points and N time outputs we have $\{\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_N)\}$ where each $\mathbf{x}(t_j) \in \mathbb{R}^M$, and we define $\mathbf{X}$ as

$$(\mathbf{X})_{ij} = x_i(t_j) - \langle x_i \rangle \tag{4.7}$$

where i indexes the grid points, j indexes the time outputs, and $\langle \mathbf{x} \rangle_i = \langle x_i \rangle$. Then $\mathbf{X}$ is an M by N matrix whose entries are time mean-centred time series of measurements at the grid points. Now instead of diagonalizing the associated covariance matrix, we instead apply the SVD. When $M \geq N$, as is common in time-indexed model output, applying the SVD to $\mathbf{X}$ we obtain [24]

$$\mathbf{X} = \mathbf{U} \begin{bmatrix} \mathbf{\Sigma} \\ \mathbf{0} \end{bmatrix} \mathbf{V}^T \tag{4.8}$$

Where $\mathbf{U}_{M\times M}$ and $\mathbf{V}_{N\times N}$ are orthogonal matrices and $\Sigma_{N\times N} = \mathrm{diag}(\sigma_1, \ldots, \sigma_N)$. The columns of $\mathbf{U}$, $\{\phi_1, \ldots, \phi_N\} \subset \mathbb{R}^M$, are the orthonormal spatial EOF basis vectors (modes), where the ith entry ϕ_{ik} in the column vector ϕ_k corresponds to the ith grid point of mode k. This basis corresponds to the singular values from $\mathbf{\Sigma}$ with

$$\sigma_1 \geq \cdots \geq \sigma_N \geq 0. \tag{4.9}$$

Carrying out the multiplication in Eq 4.8, we obtain [24]

$$\mathbf{X} = \sum_{k=1}^{r} \sigma_k \phi_k \mathbf{v}_k^T \tag{4.10}$$

where $r = \text{rank}(\mathbf{X})$. Written columnwise we have

$$\mathbf{x}(t_j) = \sum_{k=1}^{r} \sigma_k v_{jk} \phi_k + \langle \mathbf{x} \rangle \tag{4.11}$$

with the time output indexed by j. By multiplying both sides of Eq 4.8 by $\mathbf{U}^T$ we find that

$$\phi_k \cdot (\mathbf{x}(t_j) - \langle \mathbf{x} \rangle) = \sigma_k v_{jk}, \tag{4.12}$$

so that the projection of the centred data onto the EOF basis yields time-dependent coefficients defined as

$$a_k(t_j) = \sigma_k v_{jk}. \tag{4.13}$$

Therefore the columns of $\mathbf{V}$, $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\} \subset \mathbb{R}^N$, are the unscaled coefficients corresponding to each mode. The jth entry v_{jk} in the column of $\mathbf{v_k}$ corresponds to the coefficient at time j for mode k. The rank ordering of the singular values (Eq 4.9) becomes a rank ordering of the scaling of the a_k. The data can then be written as

$$\mathbf{x}(t_j) = \sum_{k=1}^{r} a_k(t_j) \phi_k + \langle \mathbf{x} \rangle \tag{4.14}$$

So that we have recovered equation 4.4, but now with the specificity that we need not sum all the way to M, but only to r. Note that there are methods of producing EOFs which are dependent on time as well as space (see section 3.2 of [20]). The covariance and SVD methods produces spatial EOFs and time dependent coefficients, which makes the interpretation of the error maps presented in section 4.3.6 and A.2.4 completely straightforward.

The submatrix of zeros in Eq 4.8 as well as the rank limited sum in Eq 4.14 both make it clear that at most the first N modes $\phi_1, \ldots, \phi_N$ are needed. This leads to the reduced SVD [62], where $\mathbf{U}$ consists of only these columns and there is no submatrix of zeros with Σ. We obtained this decomposition using MATLAB's built in `svds` command with N modes recovered to avoid the memory constraints of `svd` (see the accompanying code here and the MATLAB documentation for details). See [24] and [62] for more details on the SVD.

4.3.4 Comparison of Covariance and SVD Methods

We have just discussed two methods of obtaining the EOF decomposition from discrete data beginning in both cases with the snapshot matrix $\mathbf{X}$. We then either form the covariance matrix $\mathbf{C_X}$ and proceed as in section 4.3.1, or find the SVD of $\mathbf{X}$ and proceed as in section 4.3.3. In either case we arrive at equation 4.14, so that mathematically there is no difference, but numerically we have found that using SVD is more robust than using PCA, in agreement with [33]. This is consistent with [2] where Lemma 3.13 (pg 49) states that the SVD of a matrix is well-conditioned with respect to perturbations of its entries. So while $\mathbf{C_X}$ is real and symmetric, and so diagonalizable in principle, in practice machine precision errors may contaminate the eigenvectors. It may also be an issue with how MATLAB finds eigenvectors in each of the different cases. For these reasons we employ the SVD method in all codes, and throughout the thesis, but for completeness we will now present the mathematical equivalence of the two methods to extract a few more properties.

We have $\mathbf{X}_{M\times N}$, so there are at most $\min\{M, N\}$ singular values. Following [24], If $N \geq M$, as is common in time series data sets such as *in situ* field data,

$$\mathbf{X} = \mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^T$$

Where $\mathbf{U}_{M\times M}$ and $\mathbf{V}_{N\times N}$ are orthogonal matrices, and $\mathbf{\Sigma} = \text{diag}(\sigma_1, \ldots, \sigma_M)$ with $\sigma_1 \geq \cdots \geq \sigma_M \geq 0$. So in the field data context, we diagonalize the $M \times M$ matrix $\mathbf{C_X} = \frac{1}{N-1}\mathbf{X}\mathbf{X}^T$ as follows (dropping the $\frac{1}{N-1}$ factor):

$$\begin{aligned}
\mathbf{X}\mathbf{X}^T &= \left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)\left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)^T \\
&= \left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)\left(\mathbf{V}\begin{bmatrix}\mathbf{\Sigma} \\ \mathbf{0}\end{bmatrix}\mathbf{U}^T\right) \\
&= \mathbf{U}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma} \\ \mathbf{0}\end{bmatrix}\mathbf{U}^T \\
&= \mathbf{U}\mathbf{\Sigma}^2_{M\times M}\mathbf{U}^T
\end{aligned}$$

If instead $M \geq N$, as is common in time-indexed model output, and as was assumed in section 4.3.3, we have

$$\mathbf{X} = \mathbf{U}\begin{bmatrix}\mathbf{\Sigma} \\ \mathbf{0}\end{bmatrix}\mathbf{V}^T$$

Where $\mathbf{U}_{M\times M}$ and $\mathbf{V}_{N\times N}$ are orthogonal matrices, and $\mathbf{\Sigma} = \text{diag}(\sigma_1, \ldots, \sigma_N)$ with $\sigma_1 \geq \cdots \geq \sigma_N \geq 0$.

So for time-indexed model output, we diagonalize the $\mathbf{XX^T}$:

$$\begin{aligned}
\mathbf{X}\mathbf{X}^T &= \left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma}\\ \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)\left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma}\\ \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)^T \\
&= \left(\mathbf{U}\begin{bmatrix}\mathbf{\Sigma}\\ \mathbf{0}\end{bmatrix}\mathbf{V}^T\right)\left(\mathbf{V}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{U}^T\right) \\
&= \mathbf{U}\begin{bmatrix}\mathbf{\Sigma}\\ \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma} & \mathbf{0}\end{bmatrix}\mathbf{U}^T \\
&= \mathbf{U}\begin{bmatrix}\mathbf{\Sigma}^2_{N\times N} & \mathbf{0}_{N\times(M-N)}\\ \mathbf{0}_{(M-N)\times N} & \mathbf{0}_{(M-N)\times(M-N)}\end{bmatrix}\mathbf{U}^T
\end{aligned}$$

Which shows that the covariance matrix formed from $\mathbf{X}$ still only have $N = \min\{M, N\}$ non-zero eigenvalues, and that they match the squares of the singular values of $\mathbf{X}$. Note that the 'proof' of equality given by [33], pg 394, ignores this case. Following his abuse of notation we will also write $\mathbf{X} = \mathbf{U\Sigma V}^T$ and $\mathbf{XX^T} = \mathbf{U\Sigma^2 U^T}$ with the understanding that there may be some padding by zeros. Note that

$$\begin{aligned}
\mathbf{XX^T} &= \mathbf{U\Sigma^2 U^T} \\
\left(\mathbf{XX^T}\right)\mathbf{U} &= \mathbf{U\Sigma^2}
\end{aligned}$$

so that the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{XX^T}$. This justifies our use of the notation $\mathbf{U}$ in both section 4.3.1 and section 4.3.3. Mathematically, we could have begun with the SVD, and then using this last equation pointed out that this same $\mathbf{U}$ diagonalizes $\mathbf{XX^T}$. This means that the relationship between the eigenvalues and singular values is that $\lambda_k = \sigma_k^2$, but when $M \geq N$ at least the last $M - N$ of the λ_k are zero: the nonzero eigenvalues match the squares of the nonzero singular values (see Theorem 5.4 on page 34 of [62]).

Numerically, aside from the stability concerns mentioned, both methods produce the same reconstructions. The ranking of eigenvalues defines the order of the eigenvectors $\phi_\mathbf{i}$ in $\mathbf{U}$. We found that MATLAB codes using these methods would produce the same EOFs only up to sign. As the covariance matrix is square, this is in keeping with Theorem 4.1 of [62], which says that square matrices with distinct singular values have unique EOFs only up to sign. In data sets we expect distinct singular values, and so this result tends to hold.

This uniqueness up to sign is not relevant for the rest of the thesis for two reasons. First, we will only be using the SVD method. Second our interest is only in reconstructions, not the EOFs themselves, as was discussed in the introduction. Reconstructions are of course the same for either method. Consider equation 4.14 once more. If one ϕ_k changes sign, in order to maintain the reconstruction of the data $a_k(t_j) = \sigma_k v_{jk}$ must change sign as well. Clearly the v_{jk} change signs rather than the σ_k. So while the two methods produce the same reconstructions and singular values, the signs of the sets of EOFs and corresponding time coefficients may vary, but will always do so together.

The covariance diagonalization method of section 4.3.1 made the connection of EOFs to the dynamics of the data set clear, but in practice the SVD method is the robust numerical method we will employ. The SVD is used in all codes, and throughout the thesis. See section 15.4 of [33] for more comparison of these two methods.

4.3.5 Truncated EOF Reconstructions

Equation 4.14 makes clear that the data can be thought of as a time mean vector signal with layers of corrections provided by the EOFs. This representation recovers the data completely, so that the error in the representation of the data set is at or near machine precision. Notice that the rank ordering of the singular values (Eq 4.9) implies that each consecutive mode added to the sum contributes less variance over time than the previous

mode. To make this concrete, project the data at every time onto mode k and sum:

$$\begin{aligned}
&\sum_{j=1}^{N} |\left(\mathbf{x}(t_j) - \langle \mathbf{x} \rangle\right) \cdot \phi_k)\phi_k|^2 \\
&= \sum_{j=1}^{N} |\left(\mathbf{x}(t_j) - \langle \mathbf{x} \rangle\right) \cdot \phi_k)|^2 |\phi_k|^2 \\
&= \sum_{j=1}^{N} |a_k(t_j)|^2 \\
&= \sum_{j=1}^{N} |\sigma_k v_{jk}|^2 \\
&= \sigma_k^2 \left(\sum_{j=1}^{N} |v_{jk}|^2 \right) \\
&= \sigma_k^2
\end{aligned} \tag{4.15}$$

where we've used the fact that $\mathbf{U}$ and $\mathbf{V}$ are orthogonal, along with Eqs 4.12 and 4.13. We see that the sum over time of the contributions of ϕ_k is exactly the variance $\lambda_k = \sigma_k^2$. Note that this equation shows that the contribution λ_k from ϕ_k may be large either because of moderate contributions over most of the simulation, or large contributions over a short time, or some combination. The EOFs have been rank ordered by their total contribution to the reconstruction summed over time, but not by their contribution at any given time t_j. This time information has been summed out. This is related to the rank ordering of the singular values (Eq 4.9) providing a rank ordering in the scaling, but not a rank ordering of the values $a_1(t_j), \ldots, a_r(t_j)$ at any specific time t_j.

In order to approximate the original data set, we once again consider the SVD decomposition

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$$

and write $\mathbf{\Sigma}$ as a sum of (possibly rectangular) diagonal matrices

$\mathbf{\Sigma_k} = \text{diag}(0, \ldots 0, \sigma_k, 0, \ldots, 0)$, so that with $\text{rank}(\mathbf{X}) = r$ we have

$$\begin{aligned}
\mathbf{X} &= \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \\
\mathbf{X} &= \mathbf{U}\left(\sum_{k=1}^{r} \mathbf{\Sigma_k}\right)\mathbf{V}^T \\
\mathbf{X} &= \sum_{k=1}^{r} \mathbf{U}\mathbf{\Sigma_k}\mathbf{V}^T \\
\mathbf{X} &= \sum_{k=1}^{r} \sigma_k \phi_k \mathbf{v}_k^T
\end{aligned}$$

which is equation 4.10. This shows that the data $\mathbf{X}$ can be written as the sum of $\text{rank}(\mathbf{X})$ rank one matrices. This representation makes it clear that if the σ_i are small for some $i > D$ we can write

$$\mathbf{X} \approx \sum_{k=1}^{D} \sigma_k \phi_k \mathbf{v}_k^T \tag{4.16}$$

as a good approximation. We call this the truncated reconstruction with D modes, or the D EOF reconstruction. Clearly we take $D \in \{1, \ldots, r\}$. We consider only these rank-ordered reconstructions of all modes up to and including D, for a total of $r \leq \min\{M, N\}$ reconstructions for a given data set. The D EOF reconstruction is optimal in the sense that it is the best possible approximation among all matrices of rank up to and including D:

$$\left\| \mathbf{X} - \sum_{k=1}^{D} \sigma_k \phi_k \mathbf{v}_k^T \right\| = \inf_{\substack{\mathbf{B} \in \mathbb{R}^{M \times N} \\ \text{rank}(\mathbf{B}) \leq D}} \|\mathbf{X} - \mathbf{B}\|$$

where here the norm is either the 2-norm or the Frobenius norm. For details on this optimality see [62] chapter 5. Defining the coefficients as in equation 4.13 we can write the columnwise version of the D EOF reconstruction 4.16 as

$$\mathbf{x}(t_j) \approx \sum_{k=1}^{D} a_k(t_j)\mathbf{u}_k + \langle \mathbf{x} \rangle \tag{4.17}$$

which is a truncated version of equation 4.14.

The D-EOF reconstruction recovers as much of the original data as possible using D EOF-weighted timeseries a_k. For this reason truncated EOF reconstructions can be

thought of as an energy or variance filter, because the D EOF reconstruction captures $E = \sum_{k=1}^{D} \sigma_k^2$ of the energy, and that omitted modes correspond to omitted energy or variance contributions $\sum_{k=D+1}^{r} \sigma_k^2$. Whichever way one thinks of it, the resulting truncated series represents a simplification of the original data. If D is chosen well, this simplified data set can still capture everything of interest. However, what is of interest, and what constitutes a 'small' singular value are not universally defined, and depend on the application and the investigator. Perhaps the most intuitively clear test for defining small is comparatively, as in 'the elbow test', as described by [1], where the eigenvalues are plotted and only the modes with the largest eigenvalues are kept. Unfortunately this can bias the representation towards large scale structures, since these tend to have the most variance. Sometimes we are interested in the dynamics of small scales. Moreover modes with low energy may still represent important dynamics [47]. This is the property we exploit in section 4.3.6. Of course the elbow test cannot be applied at all if the eigenvalues decrease at an approximately uniform rate: there is no elbow test without an 'elbow.' An alternative strategy for choosing D is to apply a norm to the error of the reconstruction as compared to the raw data and choose a tolerance for error acceptable to the application. In general, the convergence of the EOF reconstructions to the data tends to make the error monotonically decrease as more modes are added. Therefore application-specific heuristic mode selection techniques based on a chosen norm and tolerance may suffice. There are more mathematically formal ways of choosing D, such as [13], which we employed in the Appendix's section A.2. However optimal choices such as these depend on the underlying mathematical framework employed. In a given application this choice of framework may be justified directly, or heuristically by its continued success for the application. A choice of mathematical framework for general application is difficult if not impossible to justify. Indeed, ad hoc choices lacking any justification besides their continual success are preferable to sophisticated mathematical methods which fail. In summary, it is not always clear how to pick D.

As an aside, we did consider the problem of picking D in heuristic terms, according to the visualizations of the reconstructions. This led to some insight, but a negative result. See section A.2 of the Appendix for details.

4.3.6 EOF Error Maps

With the background material clearly stated, we present the following novel construction. We are interested in finding features within model output fields which are worthy of further study. We will employ the SVD reconstructions just outlined to do so. As

discussed in the introduction, individual EOFs do not generally relate to individual physical processes. However, every process contributes some amount to the total variance of the model output.

Consider the following thought experiment: rank order the (unknown) processes in the dataset by variance contributed. Just as Eq 4.15 shows that the contribution of an EOF to the reconstruction may be large either as a result of moderate contributions over a long duration or large contributions over shorter durations, so too the rank ordering of processes is the result of some combination of the size and duration of each process. We expect large variance processes to include those with large scales and long duration. We expect small variance processes to include those with short scales and short duration. In between are medium variance processes with large scales and short duration, small scales and long duration, or medium scales and duration. See Fig 4.7 for examples.

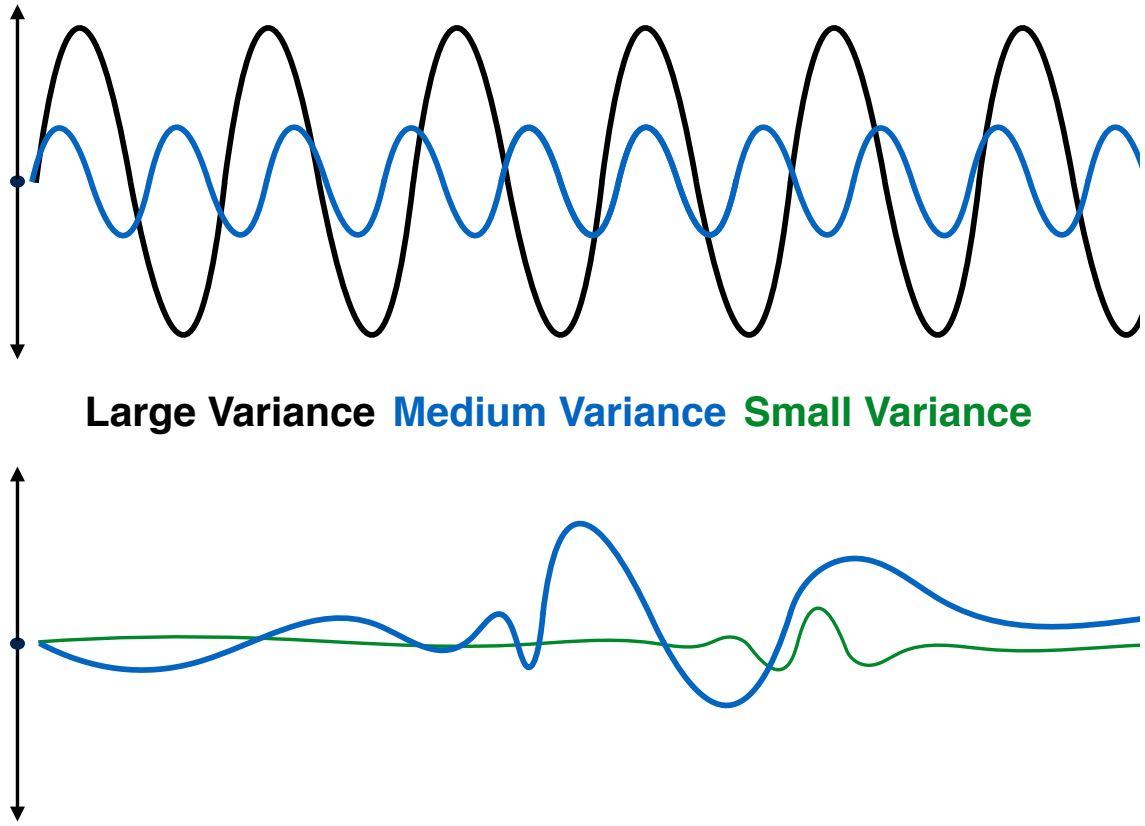

Figure 4.7: Examples of large, medium, and small variance processes over time. The upper plot shows a large variance process which has a large scale and long duration, along with a medium variance process with less variance, but equal duration. The bottom plot shows a small variance process with a small scale and short duration, along with a medium variance process with larger scale and duration.

We wish to identify time periods of interest. This means short or medium duration, and for the phenomenon to be of interest, probably medium to large scale. This means we are looking for medium variance processes in the data set, but as we have discussed, individual EOFs do not generally correspond to physical processes. Instead, the contamination phenomenon described in [71] implies that as D increases the approximations of multiple processes are simultaneously improved, and the higher the variance of a process, the greater its priority. Every mode added increases the variance represented rather than adding a process, but as variance represented increases more

processes are approximated well. By convergence, some D approximates all processes of interest. At the extreme end, if everything is of interest, $D = N$. Moreover the speed of convergence, as indicated in the scree (a plot of the normalized eigenvalues, Fig 4.10), shows that higher modes essentially represent "noise" (here the quotations are included to indicate that we do not mean noise in the sense of stochastic processes). This means that some low choice of D will tend to capture the large scales (as in the "elbow test", see [1]), while different choices of D near N are basically the same because the last modes in the decomposition have very small coefficients. Intermediate choices of D will include those that poorly approximate a variety of medium variance processes. These are exactly the processes we seek, so the error of the reconstructions can be used to find them. In particular, changes in the structure of the error over time serves as an indicator of their presence.

To better understand why reconstruction error can be used to find features, consider Fig 4.8, which reconstructions for several choices of D during the breakdown of the leading wave in the dual pycnocline data set first introduced in section 3.2. As D increases it is clear that large variance processes are approximated first, followed by smaller and smaller processes. As expected the EOF reconstruction effects multiple processes simultaneously. A choice of D near 1 corresponds to capturing processes with large variance such as the wave guide. Intermediate choices for D capture the large variance structures and some, but not all of the medium variance structures. Short to medium duration processes of interest such as the breakdown of the leading wave are poorly approximated for some intermediate D values, but as D increases the breakdown is also well approximated. Finally, a choice of D near N ($N = 100$ in this case) corresponds to an approximation which misses only noise.

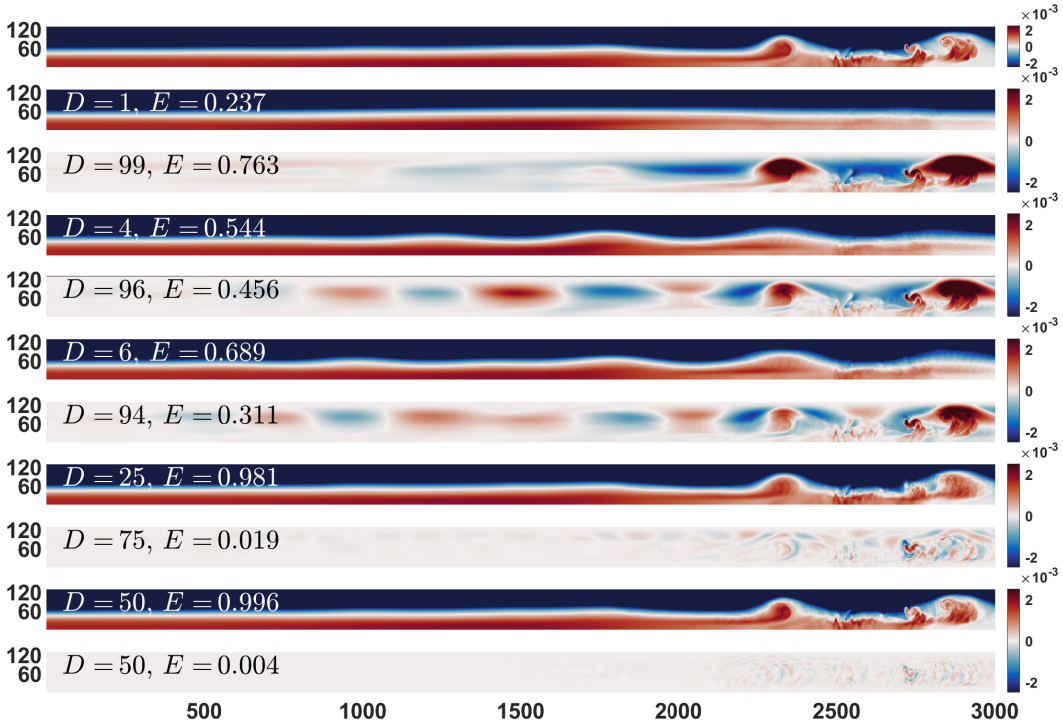

Figure 4.8: Continually increasing choices of D at time output 80 in the density field (the first 3 choices are the obvious elbow test choices). This time was chosen to look at the breakdown of the wave, which is a medium variance event with a variety of scales of structures. The top panel is the data, while pairs of reconstruction and reconstruction error are in pairs below it for comparison, with $D = 1, 4, 6, 25, 50$ increasing downward. As D increases the wave guide is approximated first, followed by lower variance structures like the breakdown, and finally the fine details of of the breakdown. By $D = 25$ the large variance wave guide is well approximated, but more modes are required to capture the fine details of the breakdown. By $D = 85$ (not shown) there is almost no error anywhere.

While Fig 4.8 shows multiple choices for D at a single time, Fig 4.9 gives an example for two different times and two different choices for D, in order to give some sense of the change in error over time for a fixed D value. For a 25 mode reconstruction the infinity norm error is greater during the shedding (bottom) than at time 20 (top). This is because for this lower number of modes, the medium variance shedding event has not yet

been fully captured. For an 85 mode reconstruction the reverse is true: the error is higher during the early time. This is because for this higher number of modes, the medium variance event has been almost fully captured, and now the very small variance structures in the early times are left (note the change in error scale between the 25 mode and 85 mode reconstructions). To summarize, then, we see that for a low number of modes the error increases during the medium variance breakdown event. This is because the larger variance background state and propagation processes have taken precedence in the reconstruction. We also see that for a high number of modes the error goes down at the time of the breakdown event. This is because there are so many modes in the reconstruction that medium variance events like the breakdown have been well approximated, and the processes that are left are virtually noise.

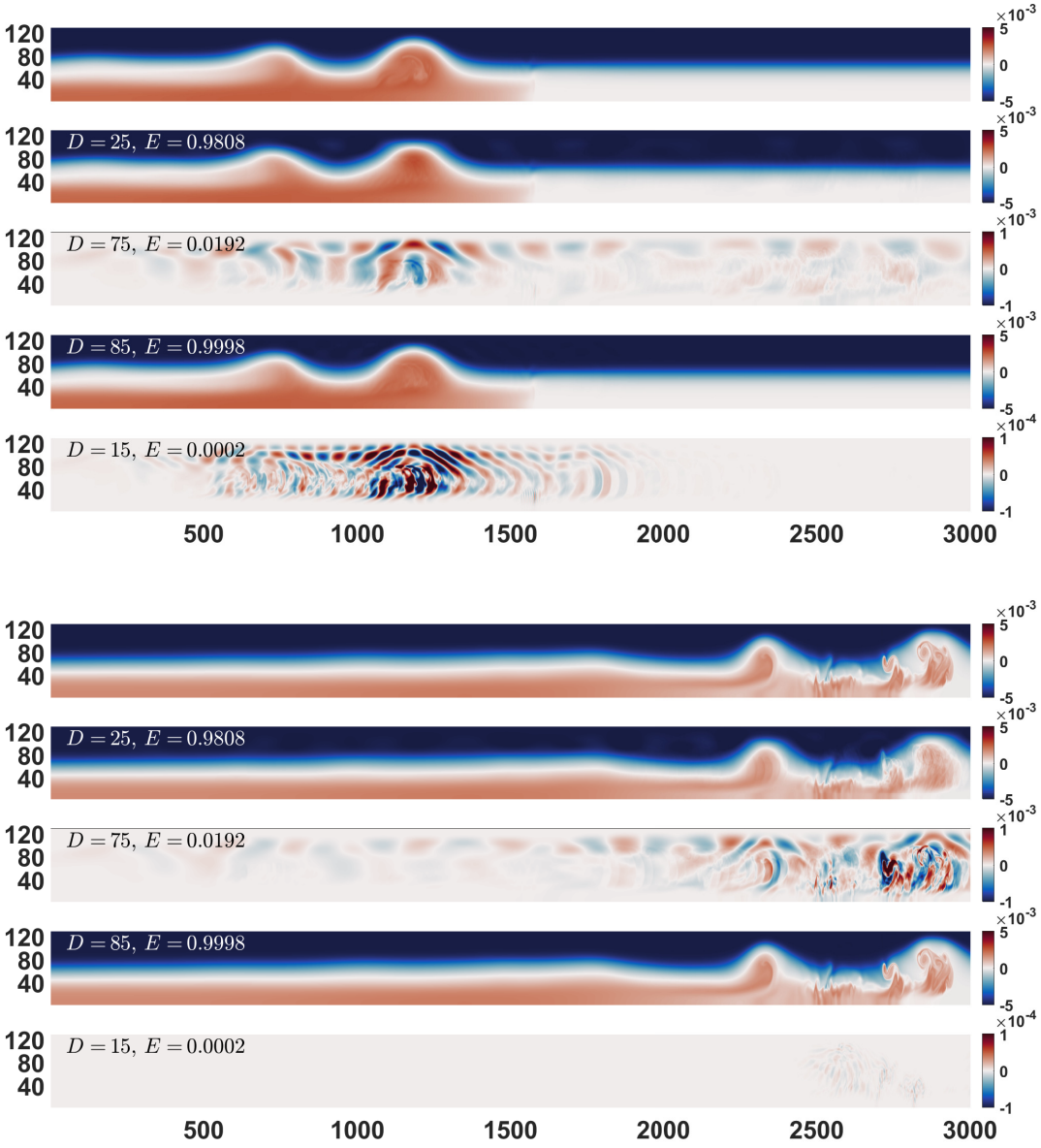

Figure 4.9: Two examples of changes in error of reconstructions over time: the upper block of panels is at time 20 and the lower block is at time 80. Similar to Fig 4.8, top panels in each block are the data, while in pairs underneath we have $D = 25, 85$ reconstructions and reconstruction errors. See text for details.

Together, Figures 4.8 and 4.9 show that the medium variance processes of interest are poorly approximated for some intermediate values of D. Since these are the processes we are interested in, we can look at the error of the reconstructions to identify when they occur. When error is high for a short time, it can indicate the presence of dynamics worthy of further study. Rather than attempt to determine a single intermediate choice for D which will help identify times of interest, we simply calculate the error of the reconstruction for every choice of D, and for all times. In order to collapse the error information to a more manageable and interpretable size, we use a norm of the time slice error, rather than a full error plot like those in Figures 4.8 and 4.9. Moreover if we use the L2 norm at every time slice the error's distribution is unknown, and may be spread thin over the whole domain or concentrated in some way. To avoid this ambiguity we use the infinity norm to make interpretation more straightforward. The error map $\epsilon_D(t_j)$ of an EOF reconstruction with D modes at time t_j is given by

$$\begin{aligned}
\epsilon_D(t_j) &= \left| \mathbf{x}(t_j) - \left(\sum_{k=1}^{D} a_k(t_j)\phi_{\mathbf{k}} + \langle \mathbf{x} \rangle \right) \right|_\infty \\
&= \left| \sum_{k=1}^{\min\{\mathrm{M,N}\}} a_k(t_j)\phi_{\mathbf{k}} + \langle \mathbf{x} \rangle - \left(\sum_{k=1}^{D} a_k(t_j)\phi_{\mathbf{k}} + \langle \mathbf{x} \rangle \right) \right|_\infty \\
&= \left| \sum_{k=1}^{\min\{\mathrm{M,N}\}} a_k(t_j)\phi_{\mathbf{k}} - \sum_{k=1}^{D} a_k(t_j)\phi_{\mathbf{k}} \right|_\infty \\
&= \left| \sum_{k=D+1}^{\min\{\mathrm{M,N}\}} a_k(t_j)\phi_{\mathbf{k}} \right|_\infty
\end{aligned} \tag{4.18}$$

for each t_j. This is simply the infinity norm of the modes excluded from a reconstruction with D modes at every time step. By construction $\epsilon_D(t_j)$ is a function of both time and the number of modes used in the reconstruction D. We call this function the error map for the EOF reconstructions of the data set, or simply "the error map." The number of modes produced by an EOF analysis is min{M,N}. The error map is therefore of size $\min\{\mathrm{M,N}\} \times N$. In the case of CFD data sets $M > N$, and so the error map has size $N \times N$. In practice, forming the error map is computationally inexpensive, as N tends to be small. The computations are simply an SVD decomposition, and one norm calculation for every time output and for every choice of D. In many contexts it is standard practice to perform an EOF analysis anyway, in which case the EOF error map is easily derived from the existing reconstructions.

4.4 Results

Although the method developed in this chapter may be applied to any time-indexed model output for which an EOF analysis would be appropriate, we will consider concrete examples from three qualitatively different simulations in stratified flow dynamics. It is not necessary that the reader have training in fluid dynamics to understand the method presented, but we provide background for each of the data sets for those who are interested. In order to keep a consistent focus, and because the varying density is the essential component of stratified flows, we will focus on the dynamics of density. As discussed in the introduction, in practice the error map method would be used to identify features in all variables within the data set. For expository purposes, we have elected to present our method on one variable in multiple flows, rather than on multiple variables in one flow.

All three data sets are from 2D simulations using a spectral collocation method (SPINS [56]). Grid doubling/halving experiments were performed to ensure that the numerical results were robust. The details of the physics of the dual pycnocline and collision cases will be discussed in future publications, while the details of the spontaneous instability case may be found in [69].

For reference, the normalized scree of the first thirty modes for all three data sets are plotted in Fig 4.10. Note that these three scree are plotted together, but that the total number of modes differs by case. The spontaneous instability data set has $M \times N = (3001 \times 156) \times 131$, so that $N = 131$ total modes, the dual pycnocline data set has $M \times N = (3001 \times 128) \times 100$, so that $N = 100$, and the collision data set has has $M \times N = (3072 \times 81) \times 150$ so that $N = 150$. The fast convergence of the eigenvalues is clear in each case. Clearly the spontaneous instability has the most variance in the first few modes, while the dual pycnocline and collision cases have more variance in higher modes.

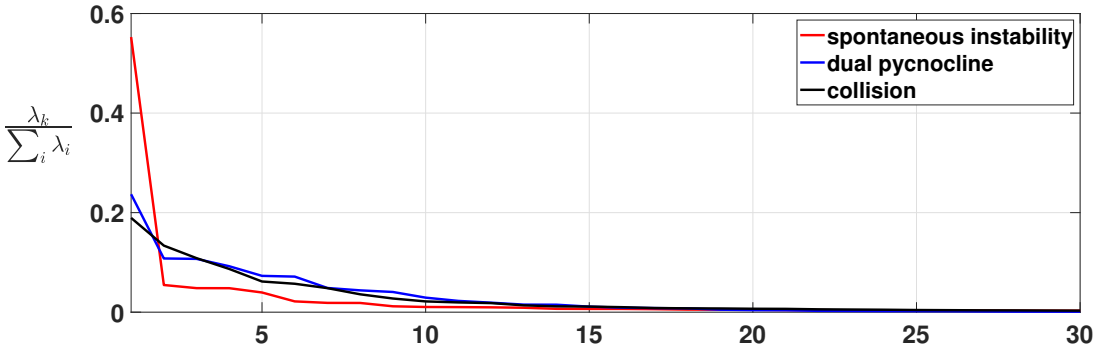

Figure 4.10: Each scree is a plot of the normalized eigenvalues as a function of mode $k = 1, \ldots, 30$, the k being the mode index from Eq 4.14. The sum in the normalization is over all eigenvalues of the given dataset. See text for details.

We now discuss the error maps $\epsilon_D(t_j)$ for each of the data sets under consideration.

4.4.1 Spontaneous Instability

The first data set is the spontaneous shear instability of a very large amplitude internal solitary wave, studied in detail in [69], following previous related work [35], [12]. Here the flow is initialized from a solution to the Dubreil–Jacotin–Long (DJL) equation, which is formally equivalent to the stratified Euler equations [54]. The initial wave develops a spontaneous instability at the rear of the wave. The instability grows and eventually exits the wave. Detailed discussion, including the effects of three-dimensionalization can be found in [69]. See the top four panels of Fig 4.11 for a visual representation of the density field's evolution in this case. The internal solitary wave serves as a "base" flow with the spontaneous shear instability playing the part of a temporary perturbation. This data set is thus close to classical hydrodynamic instability theory, for which a base flow and a perturbation are specified analytically, but still requires a full integration of the stratified Navier-Stokes equations for a full description since a purely analytical treatment is not possible in this case. In what follows this case will be referred to as the "spontaneous instability" case.

The bottom panel of Fig 4.11 shows the results of applying the error map method to the spontaneous instability data set. For times less than $t = 50$ there is very little error due to the stable background profile's large variance. This means even a reconstruction with

$D = 1$ has small error over this time period. This is consistent with the large first eigenvalue (Fig 4.10). As the instability develops, we see error in the reconstructions for small to intermediate values of D. This is due to the instability's low variance (and therefore priority) relative to the background profile, as discussed in section 4.3.6. This error continues to the end of the simulation as the instability evolves. The error map clearly indicates the presence of the instability as a time period of interest in the data set, as indicated in the obvious change in the structure of the error over time.

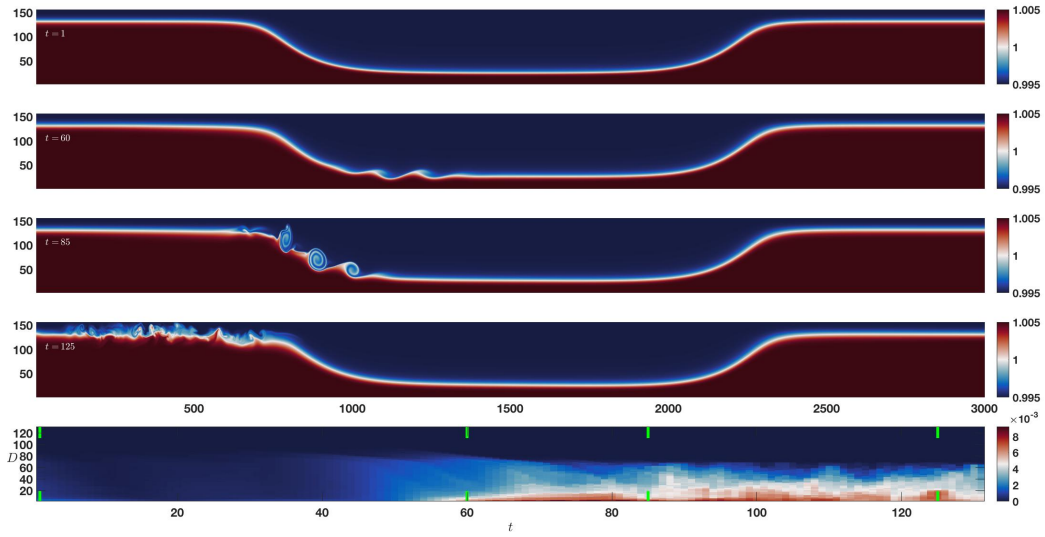

Figure 4.11: A spontaneous shear instability forms and evolves, with time increasing from the top to the bottom of the first four panels. The bottom panel is the error map with time increasing left to right, and vertical axis of increasing D, with pairs of vertical green lines indicating the times of the upper panels as time increases from left to right. See text for details.

4.4.2 Dual Pycnocline

The second data set we examine is a simulation of an internal wave train in a spatially varying wave guide, generated by what experimentalists refer to as a lock release: fluid of a set density is suddenly released from behind a barrier and is allowed to freely form waves in the stratified tank. We discussed this data set as a motivating example in section 3.2, and used it as an example in section 4.3.6. The simulation is set up so that a wave

train of internal solitary waves with a trapped core forms, propagates some distance and then encounters a sharp change in the background density (a pycnocline). This change removes the near bottom stratification, while the main wave guide remains unchanged. To the best of our knowledge, there is no *a priori* theory for the wave evolution in this cases and we find that the change in the near bottom wave guide leads to the destruction of the trapped core in the leading wave. This in turn leads to a significant increase in short length scale activity and a loss of material from the leading wave, and a significant perturbation to the second wave in the wave train. Unlike the spontaneous instability data set, in this case there is no readily apparent way to define a "base" flow in this case since even prior to the collapse of the core, the disappearance of the near boundary wave guide implies a core cannot persist [34]. See the top four panels of Fig 4.12 for a visual representation of the density field's evolution in this case. The dynamics are considerably more complex than the spontaneous instability dataset, and there is no obvious tie in with classical stability theory. This case thus acts as a stress test for our analysis method.

The bottom panel of Fig 4.12 shows the results of applying the error map method to the dual pycnocline data set. The clearest error structure is during the shedding event of the leading wave beginning around $t = 65$, up until the leading wave leaves the domain around $t = 90$. The change in structure of the error map with increasing D during this time period corresponds to the rank ordering of processes by variance illustrated in Fig 4.8 at $t = 80$. Once again the error map clearly indicates a time period of interest through the changes in the structure of the error over time.

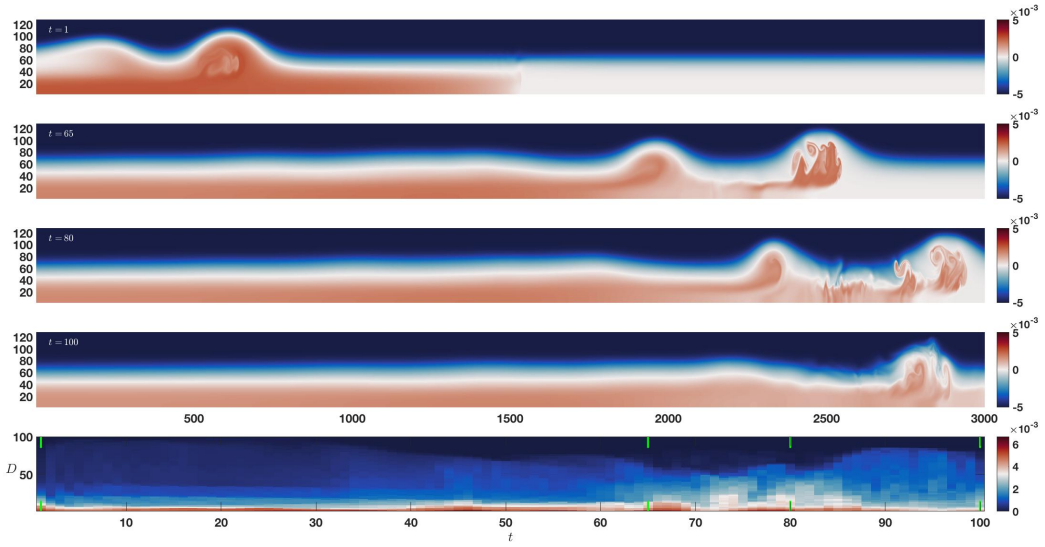

Figure 4.12: An internal wave train propagates from left to right and encounters a sharp change in the background density profile. The bottom panel is the error map with time increasing left to right, and vertical axis of increasing D, with pairs of vertical green lines indicating the times of the upper panels as time increases from left to right. See text for details.

The observant reader may have noticed the persistent error for low values of D in Fig 4.12 which was not present in Fig 4.11. The EOF modes are functions of space but not time, so propagating structures require multiple modes. This is analogous to the way a sequence of hand drawn stills can be used to create an animation, despite each picture being a functions of space only. The propagation of the basic internal waves/gravity current structure is an example of a medium scale process that lasts the duration of the simulation, requiring a minimum amount of modes to even roughly approximate. This is consistent with the scree in Fig 4.10, which shows that more variance is found in higher modes than in the spontaneous instability case. As a result there is persistent error for low choices of D even before the wave train encounters the density change around $t = 35$. This is in sharp contrast to the spontaneous instability case there was almost no propagation of the steady background state, and so even a one mode reconstruction had low error. Similarly, the slight increase in error from $t = 40$ to $t = 65$ is due to the instability in the lead wave induced by interaction with the density change. There are more small scale processes present during this time, requiring more modes to

approximate those processes well.

4.4.3 Collision

The third data set we examine involves the repeated collision of mode-1 (i.e. all lines of constant density rise and fall together) and mode-2 (i.e. lines of constant density above a given height rise, while those below fall, forming a lump-like wave) internal solitary waves in a two pycnocline stratification. This simulation is constructed based on the observations in [55] that suggest mode-mode collisions can irreversibly deform the higher mode. By choosing a double pycnocline we ensure that the interaction takes place without significant instability and three-dimensionalization. This allows us to confirm that our analysis method is capable of capturing nonlinear phenomena loosely linked to the concept of solitons, as opposed to turbulent transition. See the top four panels of Fig 4.13 for a visual representation of the density field's evolution in this case. The dynamics are complex, but compared to the spontaneous instability and dual pycnocline cases, there are no instances of short scale instabilities, and no turbulence develops. In fact, the complex pattern of constructive and destructive interference between the waves would make an analysis method based on kinetic energy or vorticity very difficult to interpret. This case thus acts as a different test for our analysis method, since the nonlinear effects in this case involve soliton–like behaviour that becomes evident during collisions (both wave–wave and wave–wall). In what follows this case will be referred to as the "collision" case.

The bottom panel of Fig 4.13 shows the results of applying the error map method to the collision data set. The waves are initialized so that the mode-2 wave is travelling rightward and the mode-1 wave is travelling leftward. As discussed for the dual pycnocline case, multiple modes are required for propagation, but in this case there is propagation of two different waves at two different speeds. This double propagation requires many modes, and again Fig 4.10 shows the variance in higher modes. The smaller error anomalies correspond to reflections from the boundary: the mode-1 wave at $t = 55$ and $t = 111$, and the mode-2 wave at $t = 141$. The large error anomaly from $t = 60$ to $t = 100$ corresponds to the overtaking of the mode one wave by the mode two wave. The clear error structure around $t = 90$ to $t = 95$ corresponds to the superposition of the two waves. As in the other two cases, we again see that the error map clearly indicates features in the data set.

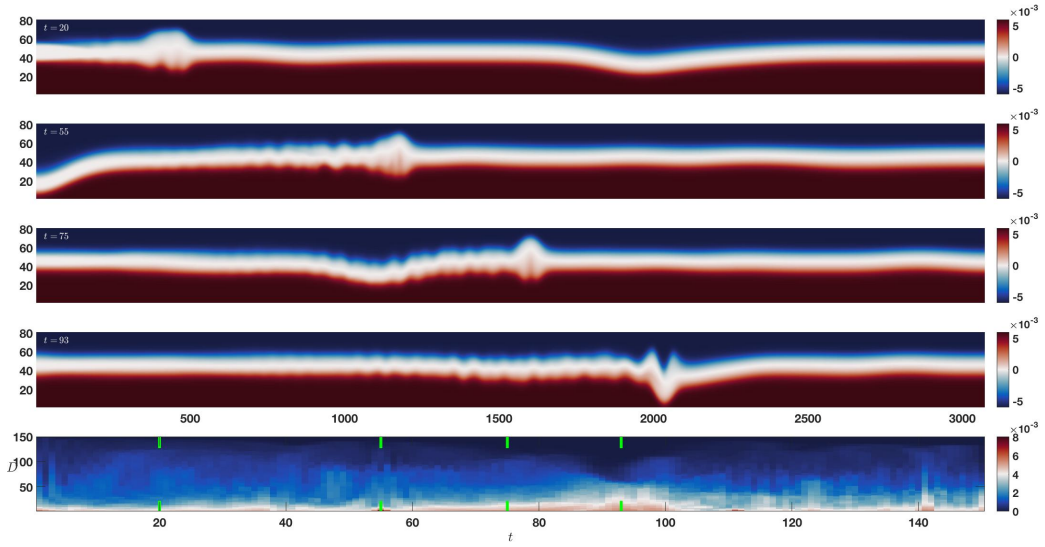

Figure 4.13: The repeated collision of a mode-1 wave with a mode-2 wave. Initially (top panel), the mode-2 wave propagates slowly from left to right, and the mode-1 wave propagates quickly from right to left. At $t = 55$ the mode-1 reflects from the left wall, as the mode-2 continues propagation to the right. At $t = 75$ the mode-1 wave has almost overtaken the mode-2 wave as both propagate to the right. At $t = 93$ the two waves nearly coincide. The bottom panel is the error map with time increasing left to right, and vertical axis of increasing D, with pairs of vertical green lines indicating the times of the upper panels as time increases from left to right. See text for details.

4.5 Discussion

The EOF error map identified time periods of interest in each of the three cases presented in section A.2.4. The method was successful even though only one of the three data sets had a classical "background–perturbation" split. And while the collision data set featured a complex patterns of constructive and destructive interference, making the kinetic energy and vorticity evolution very difficult to interpret, the error map method was still successful. Note that these two dimensional data sets were chosen so that the error map could be easily visualized alongside time outputs for expository purposes. The error map method still identifies features even if the data set is so large that it is otherwise difficult to visualize. Moreover, because the error map method collapses all

non-time dimensions for a given reconstruction and time output, the method can be applied to any time-indexed model output, provided an EOF decomposition is appropriate and computationally feasible.

For very large data sets, there are alternatives to reduce the computational burden. In particular it is clear that in many cases the full error map is unnecessary. For completeness we included reconstruction of all possible D values in the Figures of section A.2.4. Note that the error structures would have been clear with fewer modes than the maximum. In particular, for half as many modes as the maximum we could have drawn all of the same conclusions. This is unsurprising given the convergence of the eigenvalues in all cases (Fig 4.10). Of course, given the steady increase in computational power, some data sets will be too large to fit into memory. However even here, a rapidly developing literature offers a way to compute the error map, albeit with an added burden of increased computational time [68], [16]. We return to these concerns in section 6.

In the examples given here, error maps were calculated only for model output of consistent physical units. Our code [56] outputs multiple physical fields, and we chose to focus on only the density fields. As a result the EOF was carried out on a physical field with only one type of physical unit. Care must be taken if the model output includes data with different units. While multiple data types may be included together in an EOF reconstruction, the non-uniform units cause differing weights of importance on the different data types. Scalings may be chosen to attempt to correct this, but the more types of units in a data set, the more relative scalings must be considered. Moreover these scalings can have a profound effect on the resulting EOF reconstructions. All of this is a general principle when carrying out an EOF analysis. In particular, for the error map method, the relative scalings effect the reconstructions, and therefore the error maps as well. This scaling problem is most easily solved by avoiding it altogether: simply carry out a separate EOF analysis on each data type in the model output, as was done here.

The error map method has several possible extensions. For example, reconstructions from using one of the many modifications of EOF (see [17]) could be employed or a different norm chosen to measure the error. Although the focus here was on time-indexed model output, a spatial dimension could also be used as the index. In that case the method identifies spatial extents of interest, and the error map would be a function of the spatial dimension and D. In general, any dimension of a data set may be used as an index for the method, provided continuous subsets of that dimension have a useful interpretation. Such extensions are possibilities for future work.

The error map method also serves as a replacement for rough heuristics such as "the elbow test" [1] for deciding how many modes to keep. Modes with low energy, which may easily be removed by a standard elbow test, may still represent important dynamics [47]. In particular unstable modes start small but grow to be very important to the dynamics. In order to avoid missing dynamically relevant modes, simply pick a value of D large enough to avoid significant error structures in the map. This corresponds to picking the lowest row in the error map which has no significant error at any time.

EOF error maps identify features in time-indexed model output in a way which addresses the concerns of section 3.2. Thus far we have outlined the development of feature identification methods for both time series data sets, and time-indexed model output. This covers a wide variety of geophysically relevant data sets. We now consider another class of data set.

Chapter 5

Ensemble Data Sets

While previous sections have dealt with the analysis of time series data sets and time-indexed model output, we now turn to the case of ensembles of time-indexed model outputs. The gamma method is not really appropriate for these data sets for the same reasons discussed in section 3.2. Since the gamma method is easily modified it is possible that it could be applied to an ensemble data set, but we could think of no natural and non-trivial application appropriate for inclusion here. Rather than contrive a data set for the purpose of presenting an example, we will leave these explorations to future work, or to those who find the need arising naturally in their work. In contrast, there is a great deal to say regarding the application of EOF and the error map to ensemble data sets. First a digression for perspective.

5.1 EOFs and Averaging

As numericists we are primarily concerned with the discrete setting, and so the preceding sections, especially those in 4.3, developed the basic ideas of EOFs in the discrete context. Alternatively the theory can be developed on continuous functions. Historically, much of this theory was developed in the continuous setting, and it will be instructive for us to consider this viewpoint before returning to the data sets we will focus on for the rest of the section.

The main difference between the discrete and continuous derivations of the EOF is in the modelling assumptions. The continuous derivation proceeds through ensemble averaging.

Chapter 3 of [20] presents EOF (there called POD) as the solution to a variational problem. Namely choose ϕ satisfying

$$\min_{\phi} \left\langle \left| \mathbf{x} - \frac{\mathbf{x} \cdot \phi}{|\phi|^2} \phi \right|^2 \right\rangle$$

where $\mathbf{x}$ is the process and the average is taken across realizations of this process. This expression says that the mean square difference (rms error) of $\mathbf{x}$ from its projection (the vector rejection) should be as small as possible. Through some calculation of variations this leads to an eigenvalue problem with an operator whose eigenfunction-eigenvalue pairs correspond to the EOF-eigenvalue pairs.

Assuming a continuous field rather than discrete data means the analysis takes place on an infinite dimensional space. This leads to many technical concerns. For example, if the data are continuous, the integral operator in question must be compact for the algorithm to be possible. The reward for carrying out the analysis at this level of generality includes some interesting results, one of which confirms the intuition that if the eigenvalues decay fast enough then the dynamics should be on an attractor that can be approximated well. The primary goal of [20] is the construction of a simple model for coherent structures in a boundary layer. In their case EOFs are used as a target space for Galerkin projection of modified Navier-Stokes equations, and the continuous formulation serves them well. For a review of the nuances of calculating EOFs for continuous rather than discrete input, see chapter 3 of [20].

We chose the discrete derivation (section 4.3.1), which has its own concerns. It assumes that high variance for a given coordinate indicates significant dynamics, and that large cross correlation indicates redundancy. Redundancy is then reduced by diagonalization which concentrates variance along orthogonal directions. An implicit assumption made in this algorithm is that the mean and variance are enough to characterize the dynamics, but in fact only normal distributions have this property. It is also assumed that that ensemble members are sufficient in number and density to trust the resulting values of both mean and variance we obtain. These concerns pale in comparison with all the technical difficulties introduced by the continuous formulation which we have just outlined. Moreover the discrete derivation is the most practical framework to apply to the discrete data sets we wish to analyze. At this time we have no intention of developing analytic models, as we will always be performing EOF analysis on discrete data, either from CFD or the field. Besides, the discretization of the continuous problem reduces to

the SVD, as outlined in section 3.4.2 of [20]. Except for the digression in this introduction, we only consider the discrete case throughout the thesis.

One aspect of the continuous derivation raises an interesting point. If the formulation depends on an averaging operator, which averaging operator should be used? According to the continuous perspective, applying an EOF analysis as we have done so far in the discrete formulation can be thought of as treating the spatial domain as a stationary stochastic process $\mathbf{x}$ with snapshots $\mathbf{x}(\mathbf{t_j})$ acting as ensemble members, and the average taken across time. So thus far we have been averaging over time, which matches our work in section 4.3.5. One can imagine a scenario where $\mathbf{x}$ has one mean over a given time period, and then undergoes a sudden transition to another state with a different mean, over a small number of timesteps, so requiring stationarity seems reasonable. This is the view presented in section 3.2 of [20]. Note also that this view requires that we know, or can reasonably assume, that $\mathbf{x}$ is stationary before we even look at the data. How, then, could this method be applied to a dataset measuring a process about which we were ignorant?

An alternate strategy, if $\mathbf{x}$ is not known to be stationary, is to run multiple experiments, and average across the ensemble of experiments instead of across time snapshots of a single experiment. This raises further concerns. For example, what if data from only one experiment is available, as would be the case for *in situ* data sets? One option is to construct an ensemble by assuming that the underlying process is ergodic, breaking the single available record up into shorter separate experiments to form the ensemble members. However ergodicity is another strong assumption, albeit a commonly made one. If a set of multiple experiments is available, averaging across this ensemble results in EOFs which are functions of both space and time. This complication makes reconstructions less easily interpreted.

Of course many data sets have no time index at all. We will call these static data sets. EOF analyses are often carried out on static data sets under the name of Principal Component Analysis (PCA). In order to form the EOF error map for a static data set it is again required that some index in the data set serve as the parameter along which the rest of the data set is aligned. As such it is not difficult to apply the EOF error map method to a static data set so long as some dimension in the data can serve as an index. In this case features are defined with respect to this index, rather than time. For a general static data set, it is not clear that a meaningful choice of global indexing parameter would exist. Nevertheless the method could still be suitable for some data sets. For example it

would be straightforward to take a single output from a large three dimensional CFD simulation as a static data set. In the case of a gravity current with a lobe-cleft instability, the across-wave direction could serve as the parameter in this case, and we would expect the error map to show the locations of large lobes or clefts in that direction. This is a reasonable application, and indeed similar applications would be possible whenever symmetries in a simulation align with indices of the associated data set.

Clearly there are many thorny details to consider, and we will avoid them all. Following the spirit of our previous methods, we adopt a direct and easily interpreted approach. We have an ensemble of data sets under consideration, and will consider static data sets formed by taking the same time output from all experiments. This can be interpreted as the final result for the same experiment run several times. Data sets of this type have a strong relation to experimental fluid dynamics in particular, which justifies our interest in this choice. Our analysis method will be to simply average across the ensemble index i, in the same way we have been averaging across time in our previous data sets. That is, we continue from the discrete perspective, but target an ensemble of experimental results as the data set of interest.

5.1.1 EOF on a Static Data Set

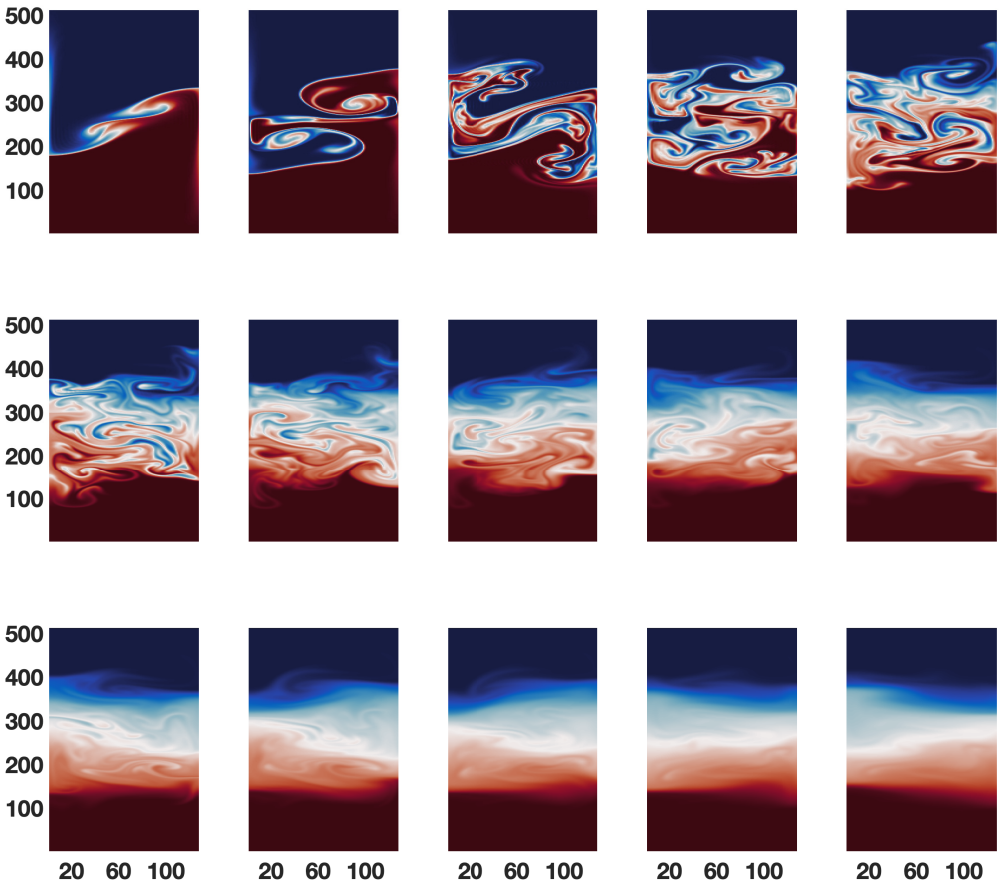

Figure 5.1: This is the density field of realization 1 of the ensemble over the 15 s run, with one panel per second increasing left to right and top to bottom. Density values go from blue to white to red as they increase.

To form the static data set, we used an experiment which can be thought of as a more energetic version of the seiche (standing wave) in a box experiment depicted in Figure 3.5 of section 3.1.2. We ran a two dimensional DNS simulation of a stratified fluid in a 128

wide by 512 tall domain. The initial conditions were nearly the same in every case, being a lightly stratified fluid with a large initial perturbation of the pycnocline. These conditions differed by a small amount of noise in each simulation. These experiments can be thought of as the experiment depicted in Figure 3.5, but with a taller and narrower domain, a weaker stratification, and a broader pycnocline initialized with a perturbation far from equilibrium. This results in dynamics dominated by the the kinetic energy of the perturbation, checked only by the weak restoring force from the stratification. The experiment was run 100 times, and the density field was chosen as the focus for analysis. Figure 5.1 shows the evolution of the density field of realization 1. After several periods of the seiche, the ensemble was formed by taking the density field at time 5 s from each experiment as the realizations of the experiment's result. We show a few ensemble members in figure 5.2. This is our static data set.

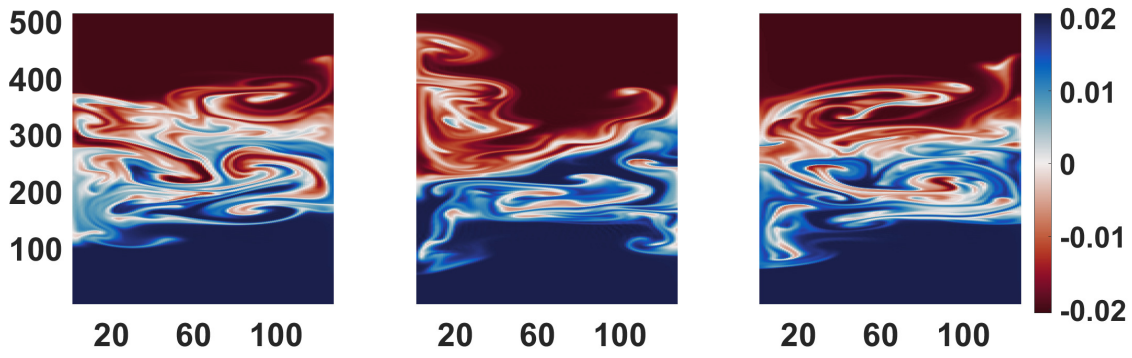

Figure 5.2: These are three realizations (1, 4, and 26 from left to right) out of the 100 in the ensemble at 5 s. Note the plume in the top left of realization 4. No other realization has this structure. Note also that the left panel of this Figure and the top right of Figure 5.1 are the same image.

As we are trying to extend the EOF error map method from dynamic to static data sets, we will compare data sets of section 4.4, and especially the dual pycnocline data set of section 4.4.2 with the static data set just constructed. Before we apply the EOF error map method, it is worth discussing the results of a traditional EOF analysis in this case. As we take the reconstructionist view throughout this thesis generally, we have not yet done this. As we will see, the error map results require some interpretation, and this traditional deconstruction of the data set into scree, modes, and coefficients facilitates this discussion. To that end, consider the normalized scree plot in the top panel of Figure

5.3. Recall that each of the data sets in section 4.4 has at least 100 EOF modes, and the top panel of Figure 5.3 only shows the first 30 of each, because the normalized eigenvalues are nearly zero by that index. In contrast, the scree of the static data set depicted in the top panel of Figure 5.4 depicts all 100 normalized eigenvalues. Note the difference in the yaxis scaling. The static data set has an almost flat scree in comparison to the dynamic cases. The first eigenvalue is much smaller, and later eigenvalues are much larger, with even the 99th normalized eigenvalue still about 0.003%. In contrast the 99th eigenvalue for the dual pycnocline data set is around 6.36×10^{-6}%. Note the 100th eigenvalue indicates convergence of the reconstruction to the original data set, up to numerical precision. So we can see that the static data set scree has much slower convergence than the dynamic data set cases.

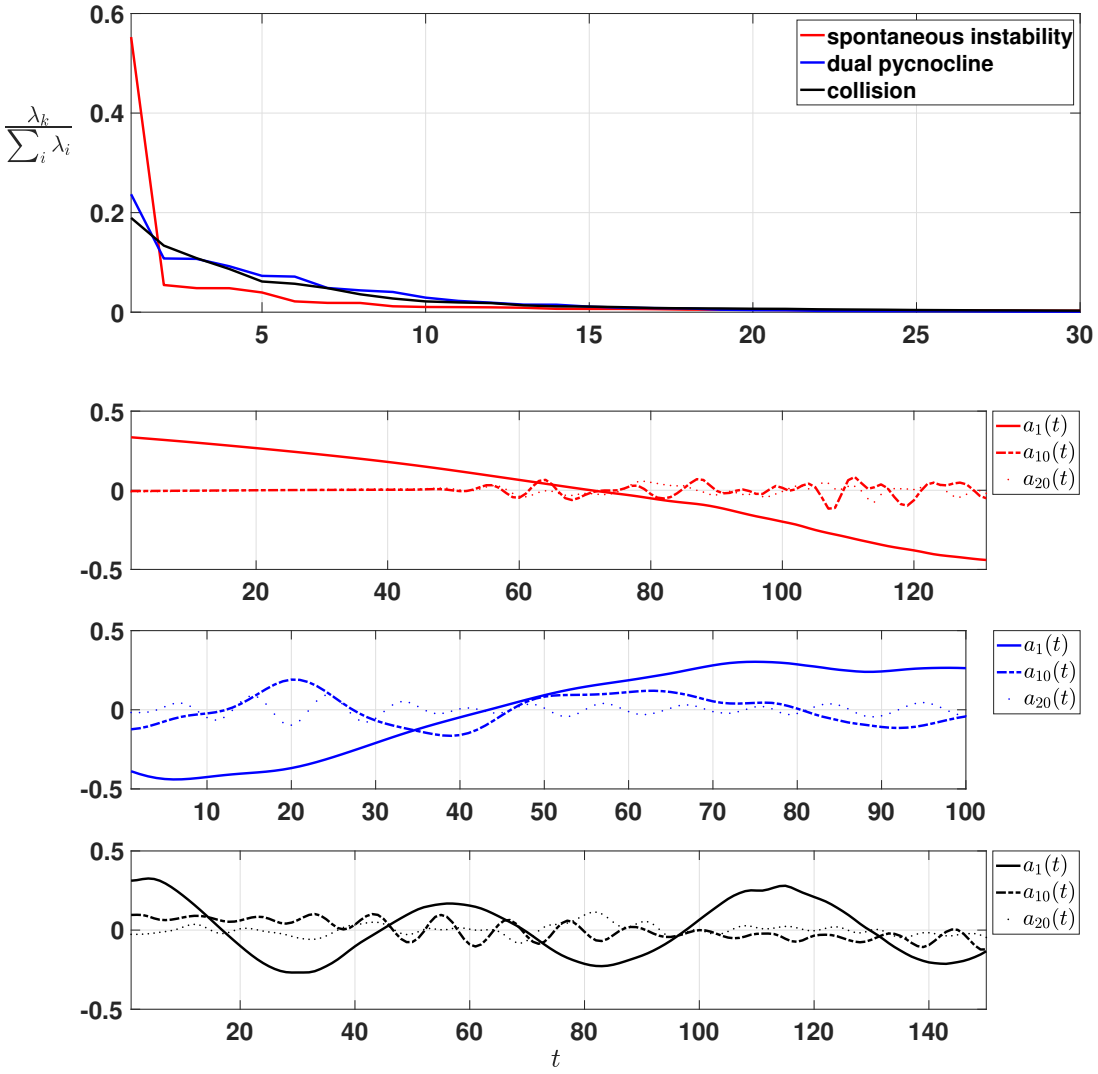

Figure 5.3: The scree and coefficient plots for the data sets from section 4.4. The top panel is Figure 4.10, repeated for ease of comparison with Figure 5.4. Each scree is a plot of the normalized eigenvalues as a function of mode $k = 1, \ldots, 30$, the k being the mode index from Eq 4.14. The sum in the normalization is over all eigenvalues of the given dataset. The bottom three panels are coefficient plots for a_1, a_{10}, and a_{20} for these three data sets as indicated by the color matching the legend in the top panel.

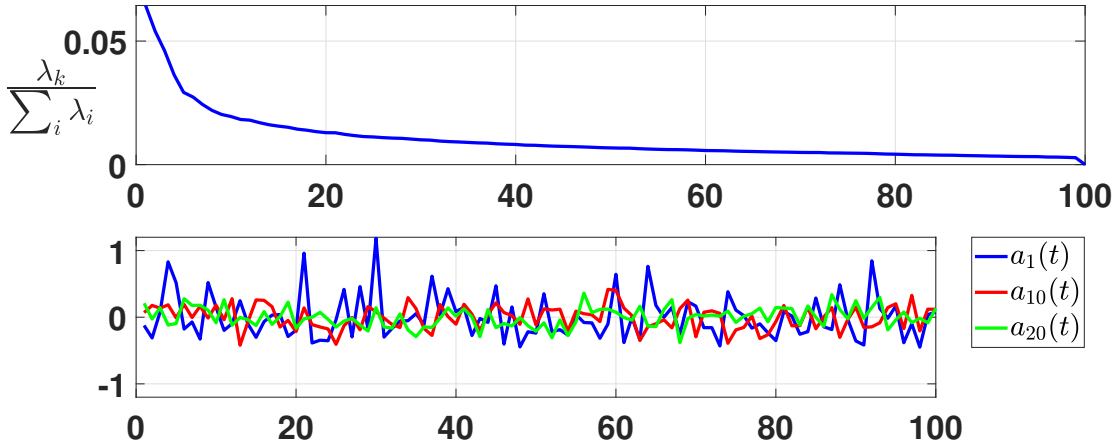

Figure 5.4: The scree (top) and coefficient plots (bottom) for the static data set. Compare with Figure5.3.

We now consider the coefficients of the EOF decomposition. The bottom three panels of Figure 5.3 show the first, tenth, and twentieth coefficients of the spontaneous instability, dual pycnocline and collision data sets, in that order from top to bottom. The rank ordering of the scaling of the coefficients is clear, as is the smooth character of their change over time t. In contrast, the bottom panel of 5.4 shows the first, tenth, and twentieth coefficients from the static data set. Notice that the scaling differences are not as clear because the eigenvalues are so similar. More importantly notice that the coefficients do not change smoothly over the ensemble member index i. So while the coefficients from the dynamic data sets change smoothly over the index of time, the coefficients from the static data set have no such smooth change over the ensemble index. This is unsurprising because each ensemble member in the dynamic data sets is a small continuous deformation of the previous one, while in the static data set each ensemble member is the end state of a different experiment, and no such smooth property should be expected in general.

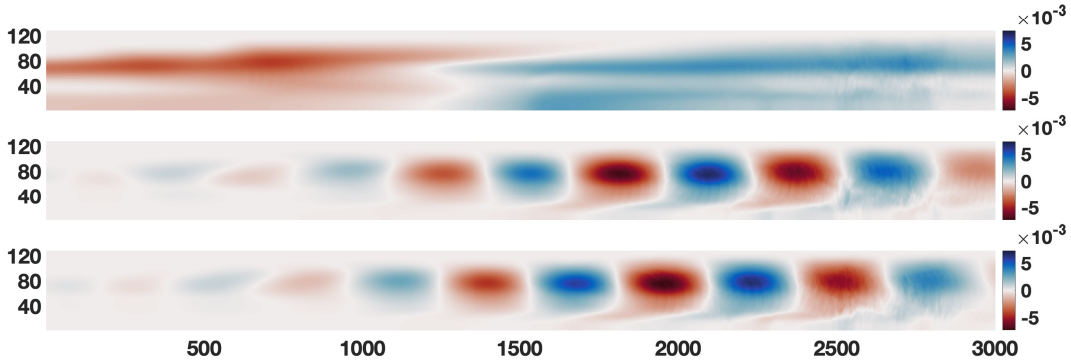

Figure 5.5: The first three EOFs of the dual pycnocline data set of section 4.4.2.

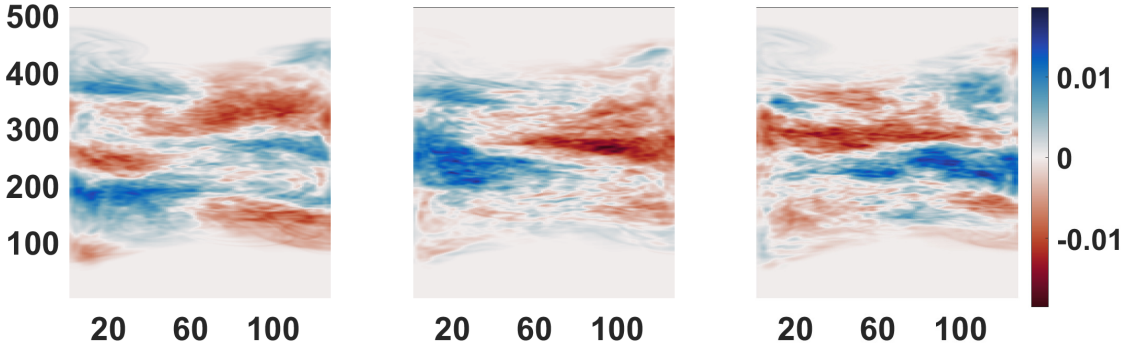

Figure 5.6: The first three EOFs of the static ensemble data set.

Finally, we consider the first three modes of the dual pycnocline data set and of the static data set. Notice in Figure 5.5 the first three modes again have a continuous physical character, and feature large scales. In particular modes two and three look like deformations of $\sin(x)\cos(y)$. In contrast, the first three modes of the static data set look 'stochastic' in nature, and feature extensive small scale structure. Once again this is due to the lack of similarity between ensemble members compared to the dynamic case.

The traditional EOF analysis being complete, we consider the reconstructions, as that is our focus and the basis for the error map. Figures 5.7 and 5.8 show the first ensemble member on the left, along with the reconstruction on the right with increasing number of

modes top to bottom. Clearly even with 50 of the 100 modes the reconstruction is quite poor. This is due to the slow convergence of the eigenvalues. With 50 modes only about 77% of the variance is represented. This is in stark contrast to the dual pycnocline data set, where as was seen in Figure 4.8 with 50 modes 99.6% of the variance was represented. In fact even the reconstruction with 90 modes has noticeable artefacts in the reconstruction for the static data set. This shows that the fast convergence of the eigenvalues is an important part of satisfactory truncated reconstructions.

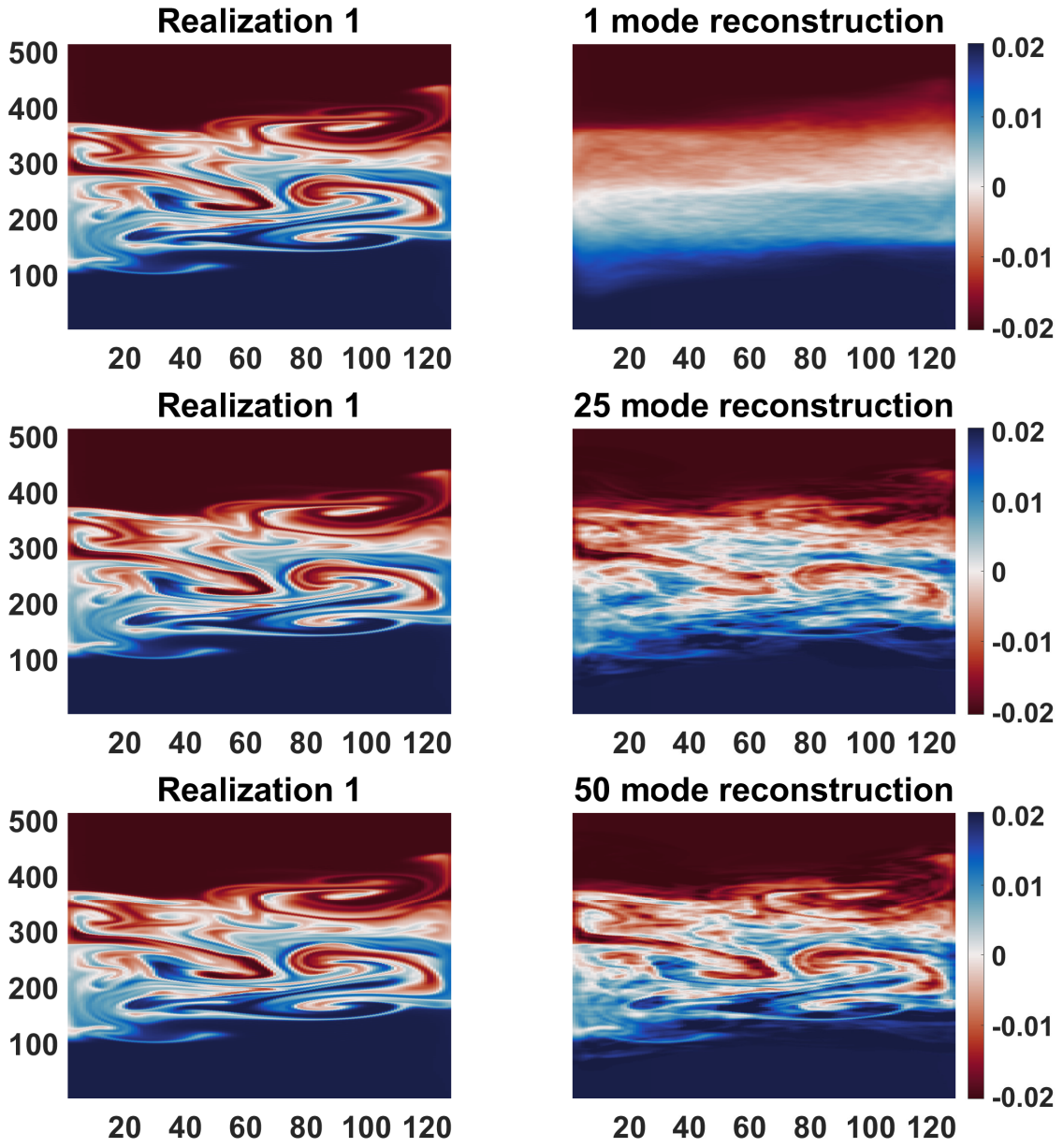

Figure 5.7: EOF Reconstructions of Realization 1. Left is realization 1, and right is a reconstruction with 1, 25, and 50 modes running top to bottom.

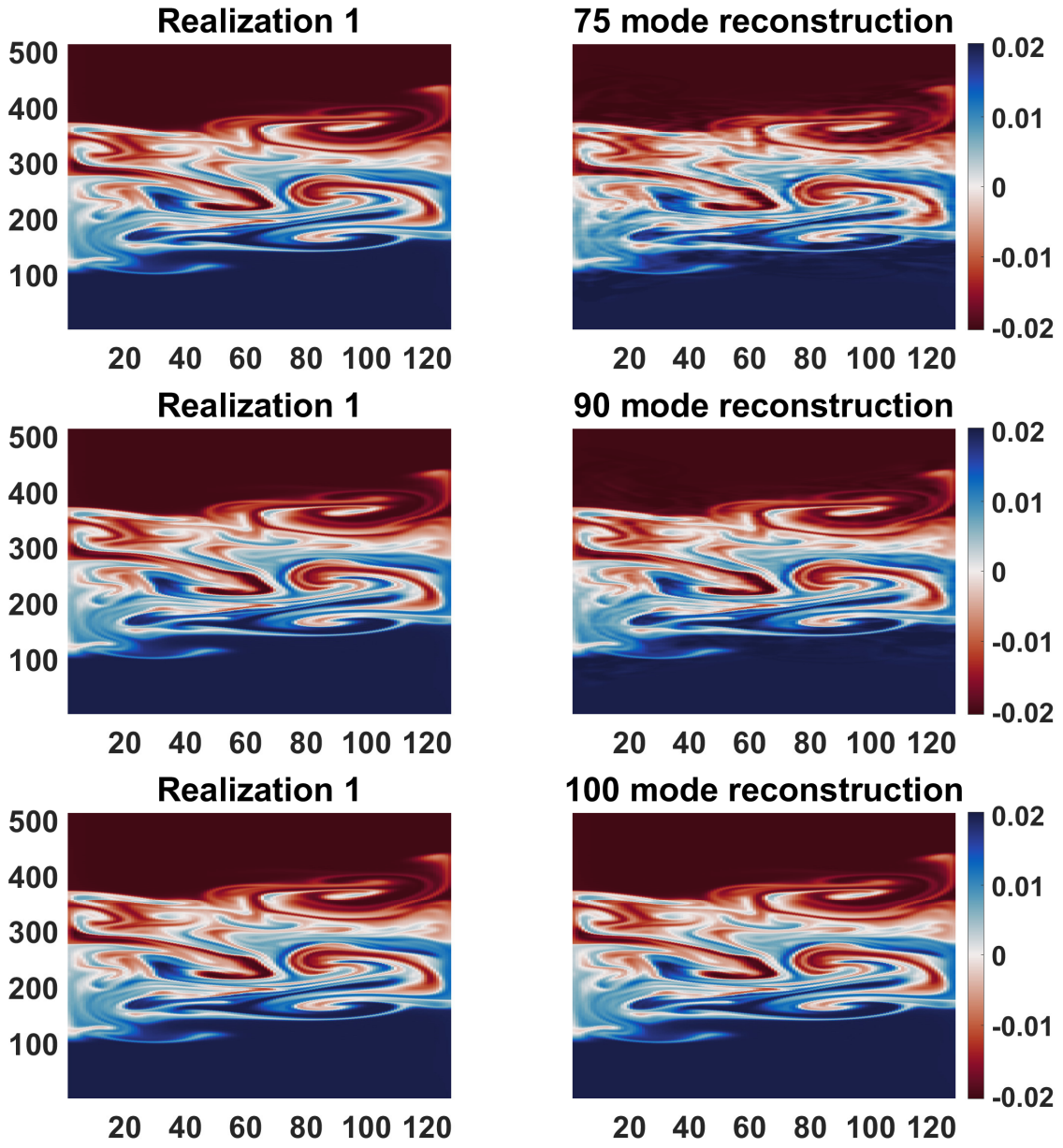

Figure 5.8: EOF Reconstructions of Realization 1. Left is realization 1, and right is a reconstruction with 75, 90, and 100 modes running top to bottom. The 100 mode reconstruction includes all modes, and so is accurate to the original data set up to numerical precision, but even at 90 modes there are still clear artefacts in the reconstruction.

In summary, we have seen that the differences across the ensemble produces very different results than that of a static data set. This is because the ensemble members have only a qualitative similarity, which yields slow convergence of the eigenvalues. This means that the reconstructions on which the error map depends require many modes to be satisfactory. Even though the ensemble members are different final states of numerical experiments with nearly identical initial conditions, the small differences in the numerics of the runs led to members different enough that there was no efficient truncated reconstruction. However each member in the ensemble looks qualitatively very similar. In contrast, most members of the dynamic data sets are shifted and deformed versions of other recent time outputs. As a result the EOF is able to capture some general trends, the convergence of the eigenvalues is swift, and the reconstructions are efficient.

Perhaps the most important point to take away from this section, is that mentioned in the discussion of Figure 5.2. Only one of the 100 realizations had a prominent plume nearing the top boundary as in the middle panel of that Figure. This raises questions of what happens if we were to run a single simulation and get this plume. One can imagine writing a paper: *"Unstable vertical transport in weakly stratified fluids"* or the like, including a detailed analysis of the dynamics. However without running an ensemble of simulations the rarity of this event would be difficult if not impossible to discuss. It could be argued that a sufficiently robust analysis of the dynamics could be enough, but no matter how convincing the argument, the ensemble of simulations immediately provides evidence. Furthermore many theoretical frameworks are built on ensembles [8], but numerical experiments using those frameworks are typically only carried out once. This is the case because of the natural constraint of clock time. Generally, a researcher would rather have a very high resolution run, than 10 or 100 low resolution runs which took the same time. Nevertheless the problem remains: if a large run yields a rare case, how likely is it to be recognized as rare. Indeed we would expect this rarity to be apparent, and indeed defined, only over an ensemble of experiments. Almost nobody does this. This may be a source of the lack of reproducibility of some results.

5.1.2 Ordered Error Maps

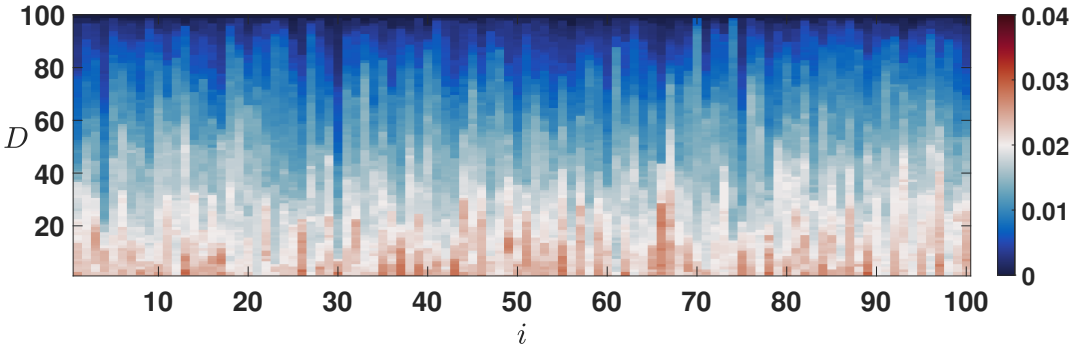

Figure 5.9: The error map results for the ensemble data set.

Figure 5.9 shows the EOF error map for the static data set used in section 5.1.1. It is clear that while it is mathematically immediate to apply the EOF error map method to static data sets, the interpretation of the error map requires further consideration.

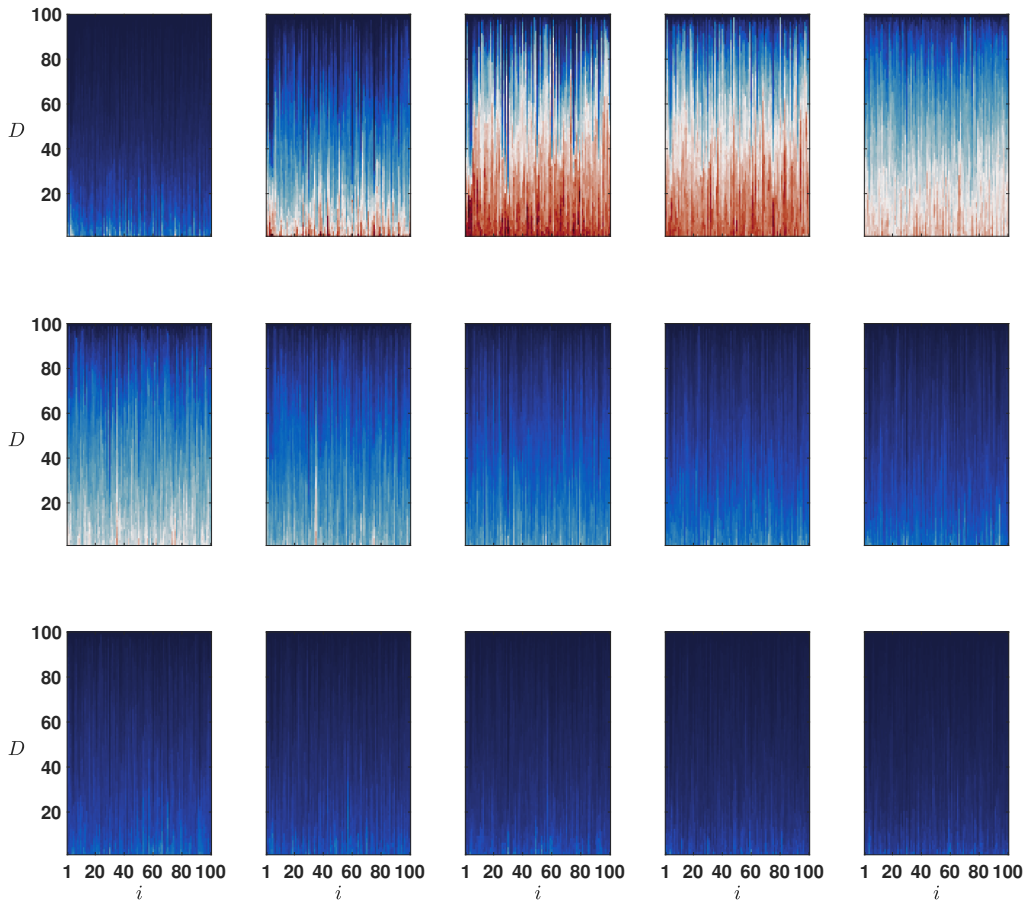

Figure 5.10: These are the error maps for every ensemble increasing in time 1 s left to right and top to bottom. Note the large errors in the maps for ensembles at 3 and 4 seconds. This is due to the large scale nature of the seiches at this time. While they are similar to each other, as shown by the scree, the small differences of the large scales lead to large reconstruction errors. The eigenvalue series of Figure 5.17 shows that the scree is flatter at time 5, corresponding to more differences across the ensemble, but less reconstruction error.

Figure 5.10 shows that the error maps for static data sets formed at other times are just

as difficult to interpret, which shows that it was not just the selection of the ensemble at 5 s which caused the problem, but the lack of a physically meaningful ordering. For dynamic data sets time provides an ordering of the outputs. The error map for a dynamic data set depicts error change over time as a way to identify time periods of interest. In an ensemble such as the ones we've constructed, there is no clear ordering. As a result the index i simply enumerates the order we chose, and there is no meaning to a 'period' over some sub-range of i which forms a feature. This makes it impossible to interpret the error map for static data sets of this type in the same way as we would for dynamic data sets.

This is not a mathematical problem, but a problem of interpretation only. The covariance matrix from equation 4.3 will have the same entries if the times t_k are re-ordered. Put another way, the covariance matrix is unchanged if a time-indexed set of vectors is put in any order, not just that of increasing time. In a static data set, the vectors have no natural ordering, but any order may be chosen without effecting the covariance matrix. This means that the EOFs of the covariance matrix are also unchanged by re-ordering, and we may chose any order without concern that it will change the result of the EOF analysis. Since the index i enumerates the order we happened to chose, and this ordering does not change the resulting EOF analysis, it does not effect the reconstructions or their error. Therefore the columns in the error map can be re-ordered in any way without losing information. This is in stark contrast to the dynamic case, where the time-ordering is essential.

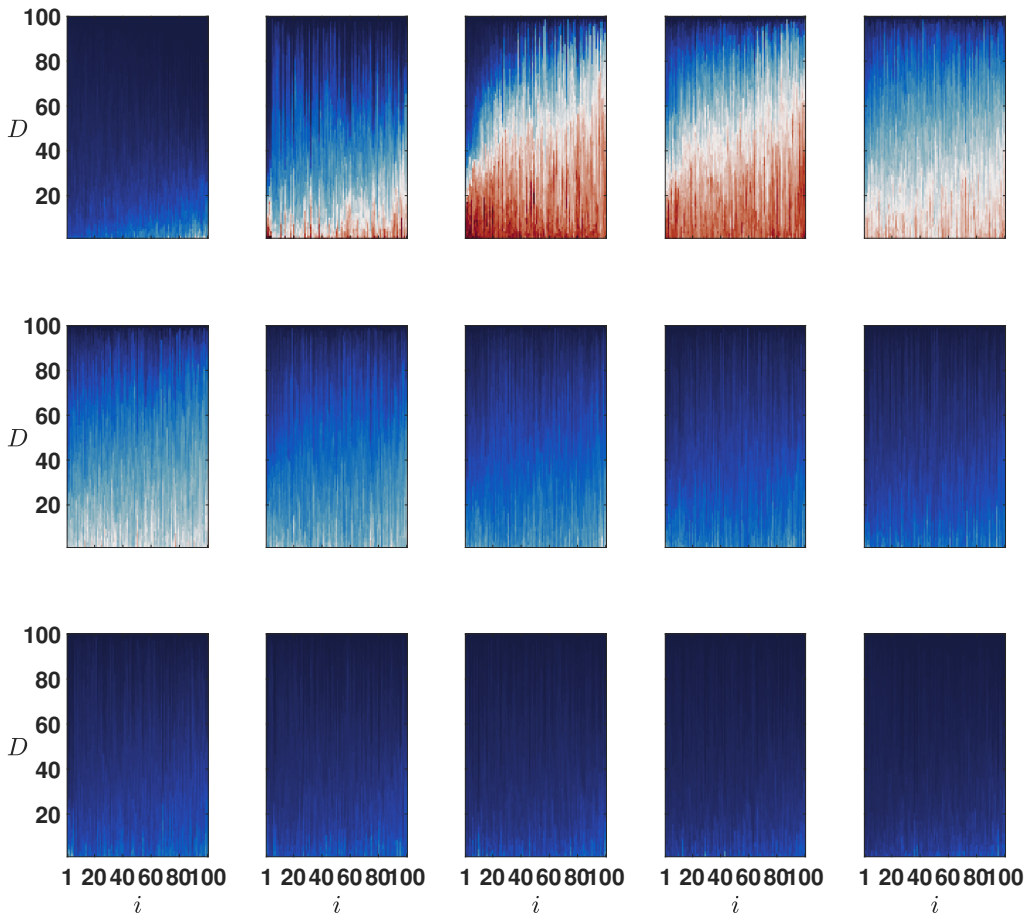

Figure 5.11: This is the same as Figure 5.10, except that each error map has been ordered by total reconstruction error over D. See text for details.

While there is no clear ordering on a static data set, the error map can be used to construct one. To form a new ordering we need a criterion. An obvious choice is by total error across all reconstructions. Mathematically, this corresponds to summing the errors along each column of the error map. The results of this procedure are depicted in Figure 5.11. In each panel the realization on the left of each error map is the one with the least

total reconstruction error over all values of D, and the one on the right has the most error. One could interpret this to say that the realization having the least reconstruction error is the most representative of the ensemble, and that having the most reconstruction error is the least representative of the data set. We see that most of the ensembles have low enough error that the ordering is not useful. As seen in Figure 5.1 times 2 through 5 have significant error caused by variations in stirring and mixing across the ensembles. The ordered error maps show that only times 3 and 4 have significant differences in error across the ensemble, while the error across the ensembles in outputs 2 and 5 are relatively flat.

We will consider an example from the static data set taken across the ensemble at time 3 s. Figure 5.12 shows the evolution of realization 30, which had the least error in the ensemble at 3 s. Figure 5.13 depicts the evolution of realization 4, which had the second least error at that time, and Figure 5.14 depicts the evolution of realization 28, which had the third least error at that time. Comparison of these Figures at time 3 shows strong similarities. Finally, Figure 5.15 depicts the evolution of realization 59, which had the most reconstruction error at time 3 s. There are clear differences between this realization and realizations 30, 4, and 28 at time 3 s. The entire evolution for each realization was included for a broader comparison, but we remind the reader that the ranking was according to reconstruction error only at the 3 s ensemble.

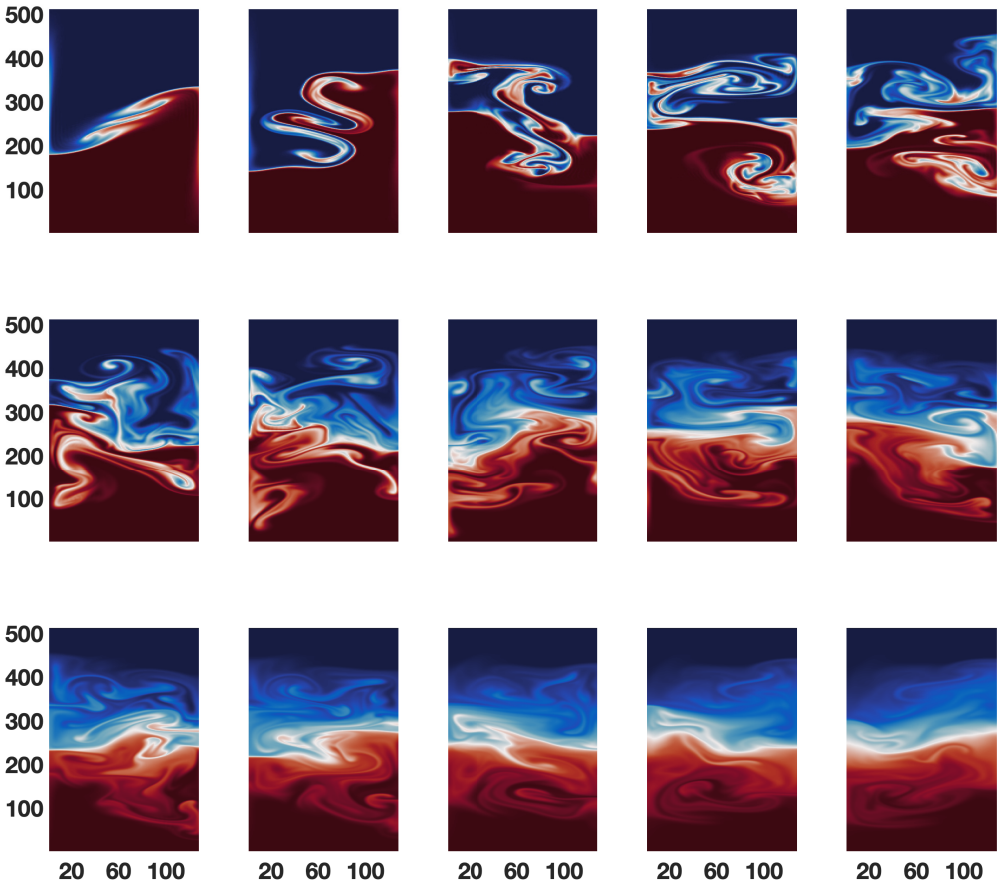

Figure 5.12: This is the density field of the 30th realization over the 15 s run, with one panel per second increasing left to right and top to bottom. The top middle panel therefore corresponds to time 3 s. This realization had the lowest error at 3 s.

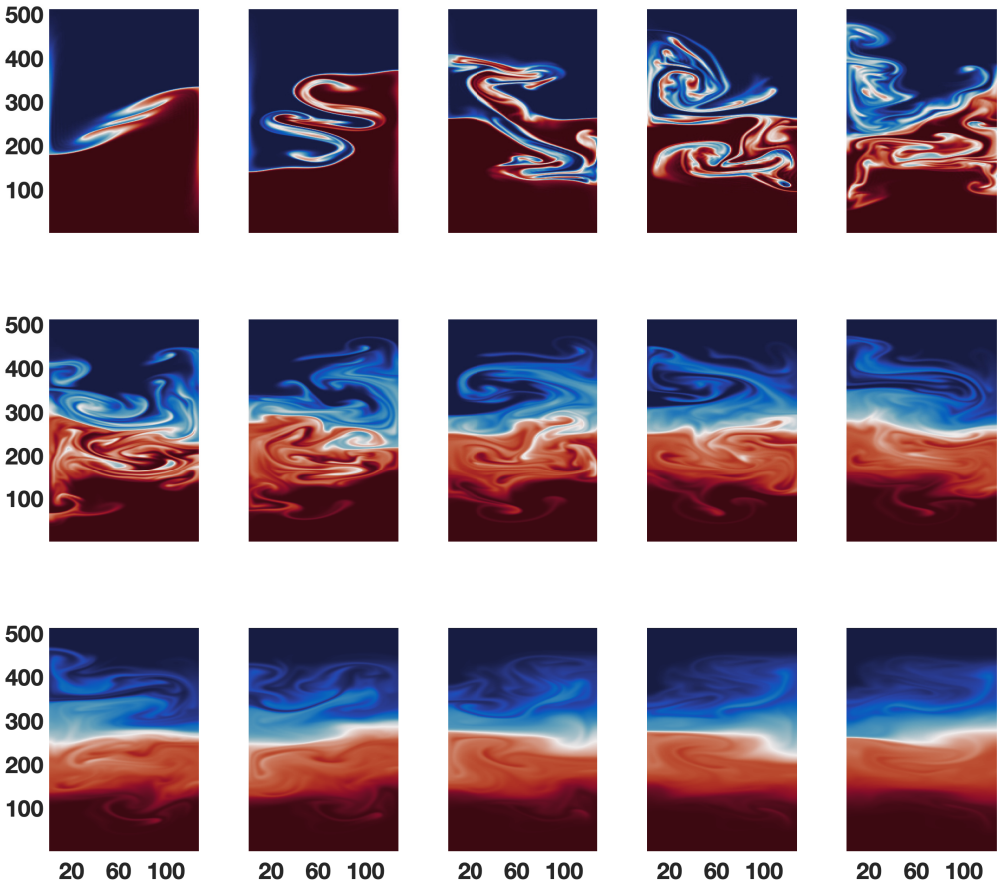

Figure 5.13: This is the density field of the 4th realization over the 15 s run, with one panel per second increasing left to right and top to bottom. While realization 30 had the lowest error at 3 s, realization 4 had the second lowest at 3 s. The entire run is included for comparison, but the comparison at 3 s shows clear similarities to realization 30.

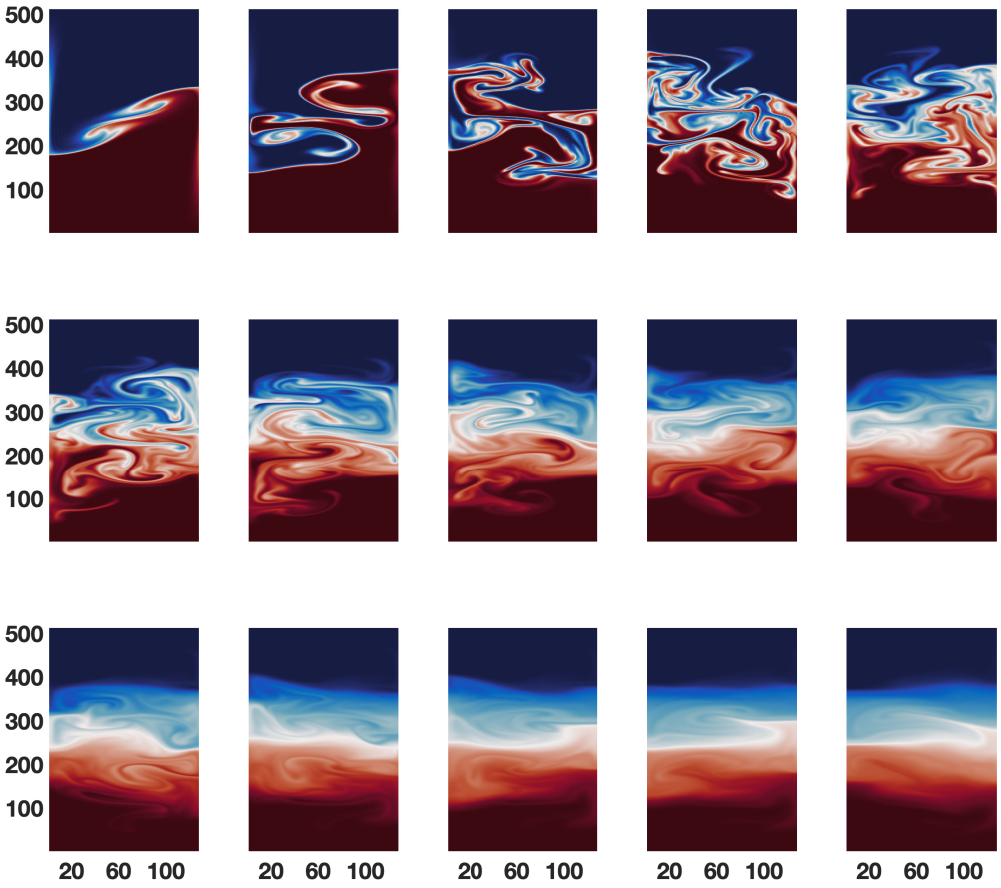

Figure 5.14: This is the density field of the 28th realization over the 15 s run, with one panel per second increasing left to right and top to bottom. Realization 28 had the third lowest error at 3 s. The entire run is included for comparison, but the comparison at 3 s shows clear similarities to realizations 30 and 4.

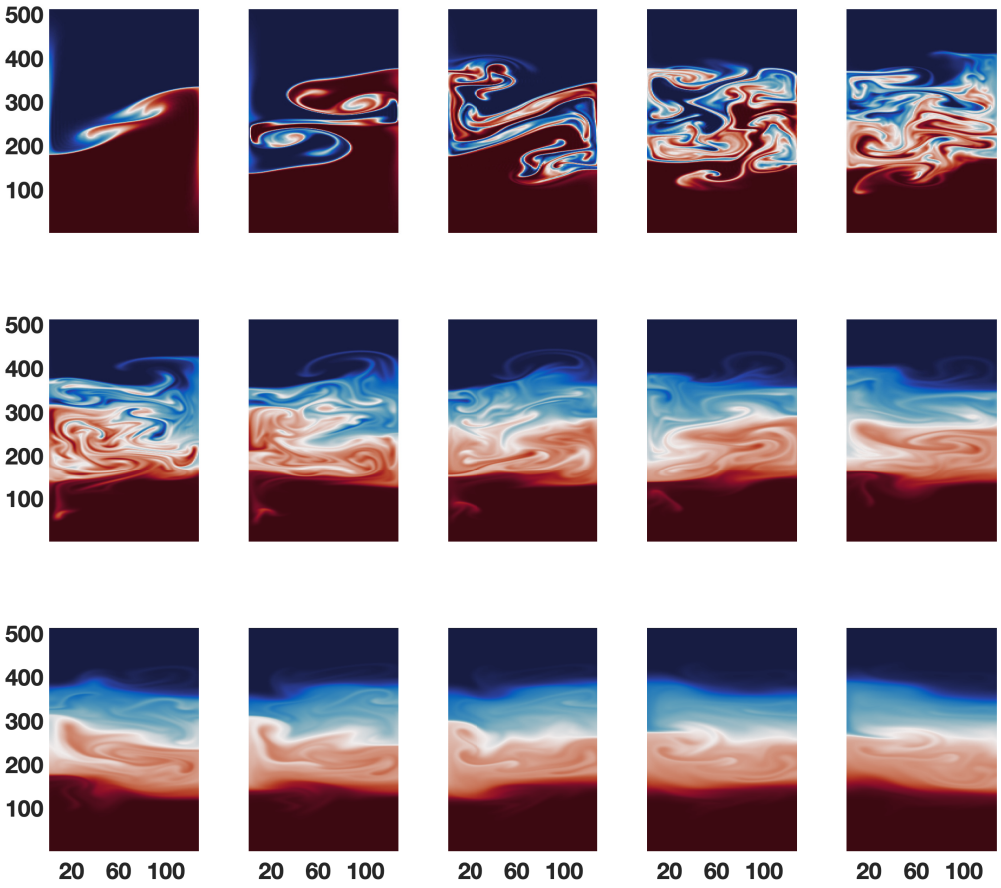

Figure 5.15: This is the density field of the 59th realization over the 15 s run, with one panel per second increasing left to right and top to bottom. While realization 30 had the lowest error at 3 s, realization 59 had the most error at 3 s. The entire run is included for comparison, but the comparison at 3 s shows clear differences between this run and realizations 30, 4, and 28.

It seems clear, then, that ordering by total reconstruction error is an ordering of how representative a realization is of the ensemble, with the most representative realization having the least error and the least representative realization having the most error. As

the EOF analysis is performed on the perturbation from the mean, in this way an ensemble is ordered by it's distance from the mean. This gives a way to identify both realizations which should be studied as being representative of the underlying processes, and those which are in some sense outliers.

The EOF error map was developed for use on data sets indexed by time. It is therefore not surprising that it required modification for use on static data sets. In particular our application of the EOF error map method to static data sets has been hindered by the lack of a natural ordering of the data sets. Ordering by total reconstruction error was our solution to this problem, and the ordered error maps are more useful than the unordered error maps because sub-ranges of i have meaning: sub-ranges with no significant change in the error structure correspond to similarly representative realizations. This is an ensemble data set version of a feature. This principle can be seen in the static data set at time 3, where the first few realizations correspond to similar error in the ordered error map, and similar structures in the Figures just discussed. We would need more testing to conclude anything general with confidence, but one can imagine a scenario where the error structure indicates a bifurcation or other significant event. Depending on context it is possible that other criteria would be useful. Perhaps the error of a particular reconstruction, rather than the sum. Perhaps an iterated model where certain sub-ranges of i are kept, and the EOF run again on the members of this subset. This is a definite direction for future work

It should be noted that for some reason realization 30 had the smallest summed reconstruction error not just at 3 s, but in fact in 9 of the 15 time outputs: times $t = 2, 3, 5, 6, 7, 8, 9, 10, 12$. By extension of the least error at every time, this makes this realization in some sense representative of the ensembles over time as well. This could be a useful metric for ensembles of experiments, but there is no reason to think that there will always be a realization which has this property. This is another avenue for future work.

5.2 First Eigenvalue Series

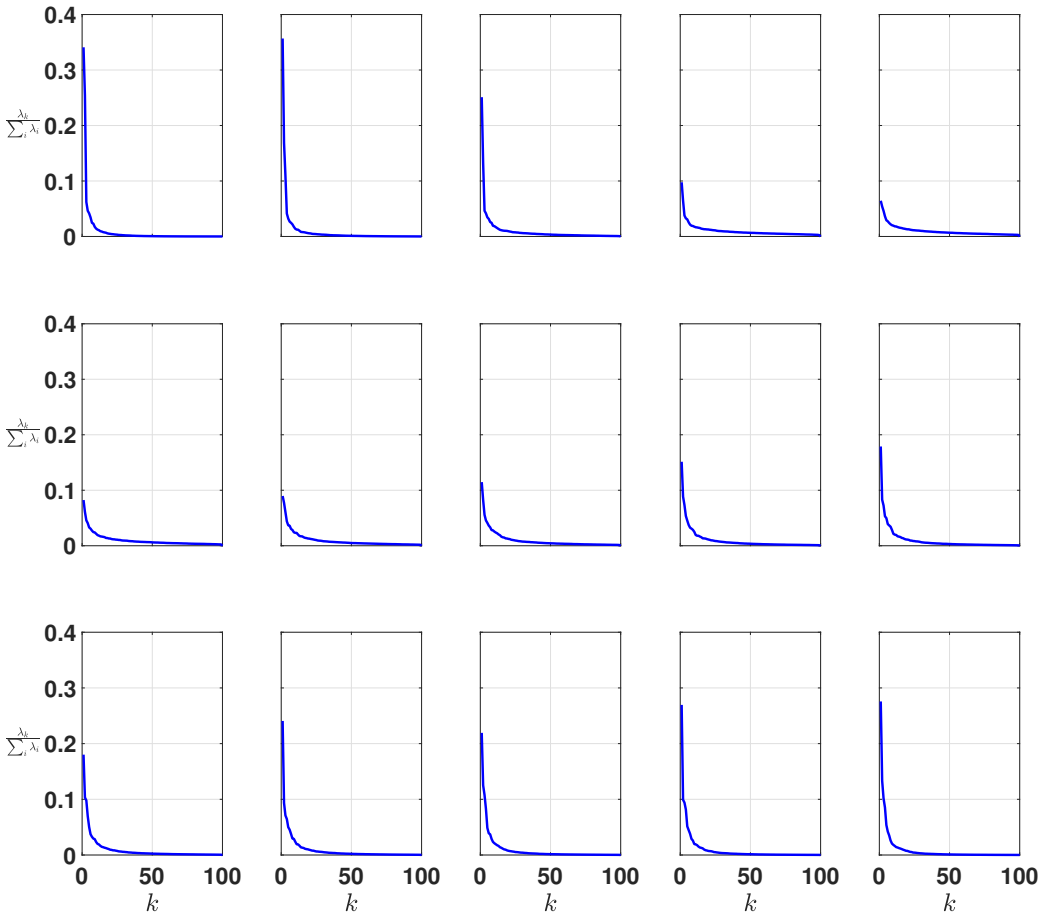

Figure 5.16: The normalized scree for ensemble data sets formed at 1 to 15 s, in order from left to right and top to bottom. This makes the top right panel the same scree as that in the top panel of Figure 5.4. Note that this time is associated with the slowest convergence of the eigenvalues.

Having discussed EOF and error maps on static data sets, we now consider the full ensemble of 100 experiments over 15 different time outputs. We again derive a method for identifying times of interest. Note that in Figure 5.11 time 5 has the flattest error map, while still having significant error at all times. Compare with Figure 5.16, which shows the normalized scree results from the static data sets formed by taking the realizations at 1 through 15 s inclusive. Our choice of 5 s in section 5.1.1 was the special case of the ensemble with the most slowly converging scree of the 15 choices. Note that for early and late times the normalized first eigenvalue is similar to the dynamic data sets whose scree are plotted in the top panel of Figure 5.3, indicating much more agreement across the ensemble. The reason for this is that each experiment starts as a seiche, then breaks down, and then settles to an available potential energy (APE) minimum. Near the beginning of the experiment all realizations represent a very similar seiche, and near the end all realizations represent a similar relatively quiescent and stable configuration. It is the details of the turbulent transition which causes the considerable difference across the ensemble and the relatively flat scree. Out of the 15 times, this transition is most prominent at 5 s. Put another way, the large scale physical similarity across the ensemble at early and late times yields better convergence than the small scale differences during the turbulent breakdown. The differences between realizations are small, as in the dynamic data set case. In the dynamic case the differences are small because the evolution of the process leads to small differences between time outputs. In the early and late time static cases the differences are small because the physics is better determined outside of turbulent times.

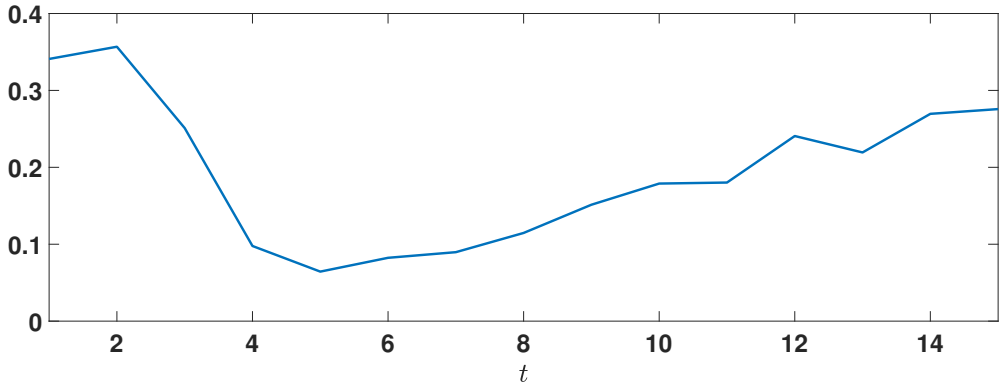

Figure 5.17: The first eigenvalue of each panel of Figure 5.16 as a line plot. We call this a first eigenvalue series. Note that the minimum occurs at $t = 5$, which is why we took this ensemble to form the static data set studied in section 5.1.1.

These observations suggest the following algorithm for finding interesting times in ensembles of experiments. Note that by normalization, the first eigenvalue serves as a proxy for the entire normalized scree. A lower first eigenvalue corresponds to slower convergence and more disagreement between realizations. A higher first eigenvalue corresponds to faster convergence and more agreement between realizations. We can therefore take the first eigenvalue from each scree in Figure 5.16 to form a series, and the minima of that series indicates a time of disagreement across the ensemble. Formally, given a set of L experiments of size $M \times N$ (M corresponding to number of grid points, N corresponding to number of time steps) with $M > L$. Take the ensemble of grid points M across the L experiments so that the SVD is performed on the resulting $M \times L$ matrix, at every time t_j, $\jmath = 1, \ldots, N$. Extract the EOF scree and normalize. Save the first normalized eigenvalue at each time t_j to form the time series $\lambda(t_j)$. Then minima of $\lambda(t_j)$ correspond to times of disagreement across the ensembles. The result of this algorithm applied to the 15 ensembles under consideration is depicted in Figure 5.17. This first eigenvalue test justifies our choice of the ensemble at 5 s when forming the static data set.

5.3 Discussion

In this chapter we continued the extension of feature identification methods to ensemble data sets. We saw that the unordered nature of these data sets led to problems of interpretation, most notably in what was meant by a feature in this case. The imposition of an order by total error solved this. Features are sub-ranges with no significant change in the error structure. These correspond to similarly representative realizations. Moreover individual realizations are ranked by how representative they are of the ensemble. In this way the ordered error maps for static data sets give meaning to single ensemble members in a way that error maps on dynamic data sets do not.

Our static data sets were constructed from single time outputs of a set of dynamic experiments. The first eigenvalue series served as yet another feature identification method in the dynamic ensemble data set case. Along with the gamma, error map, and ordered error map methods, we now have feature identification methods for an enormous range of data sets.

Chapter 6

Extending the Data Pipeline

For the most part, our data sets have been small enough to visualize in a few panels of a single figure. In this sense previous examples of the application of our methods have been at tutorial scales. This has meant that visualization has preceded analysis in every case. Now that the methods have been established in our minds through previous sections, we will apply them to full size data sets. Both methods show their practical worth in being applied before the visualization as a way of focusing further analysis efforts, including any visualization. In this way the methods serve as a primary method to identify features, rather than as confirmation after visualization. As mentioned, SPINS [56] outputs high resolution bulk measure time series. The gamma method scales without difficulty, and can be applied to any data set SPINS may produce. It is less straightforward to apply the error map method. We outline this problem and our solution now.

6.1 EOF on Large Data Sets

The data sets in sections 2 and 4 were small enough that built in MATLAB functions were always sufficient to perform the calculations in RAM. This is not always the case. Data from DNS of fluid dynamics problems is large by design. In fact, according to the categorizations of [18] listed in their Table 1 (which in turn is quoted from [21] and then extended), our large runs belong in the 'monster' data category ($\sim 10^{12}$ bytes). For example, a SPINS run of 1024^3 and $N = 100$ time outputs produces four scalar fields: density and the three components of velocity. Each number stored is 8 bytes. Therefore the total number of bytes is

$$1024^3 \cdot 100 \cdot 4 \cdot 8 \approx 3.4 \times 10^{12}$$

Using these numbers $M = 4 \cdot 1024^3$ and computing EOFs as described in sections 4.3.1 requires diagonalizing the $M \times M$ dense matrix $\mathbf{C_X}$ of equation 4.3. Alternately, we may compute the SVD as discussed in section 4.3.3 on a large and dense $M \times N$ matrix $\mathbf{X}$. Both methods are extremely resource intensive, and tend to exceed available RAM. Fortunately, MATLAB's `svds` command has an option to supply your own multiplication operation. We constructed an operation which writes the matrix rows and columns to file, and then reads in only what is needed for a given calculation. This is more time consuming, but makes the calculations possible. It is a simple matter to then construct the error map.

6.2 Results

6.2.1 Cabbeling In a Stratified Shear Instability

The first example is a simulation of stratified shear instability on the lab scale, in the cold water regime. Cold water is interesting because the density is a non-monotonic function of temperature, so that there is a temperature at which maximum density occurs (around 4 degree Centigrade for pure water). This is why ice floats. We are interested in shear instability in this setting because when water is cold it is possible to mix two parcels of fluid that have different temperatures, but the same density, so that when mixed the resulting parcels will have a larger density than either of the two had at the outset. This phenomenon is called "cabbeling" [3]. It is challenging to model numerically due to the small scale associated with instability onset. In our simulation that dimensions of the rectangular tank are $(0.256, 0.064, 0.128)$ m with $(512, 128, 256)$ grid points, implying a resolution of 0.5 mm in all directions. Outputs are 5 s apart, and the simulation is stopped after 48 outputs. The simulation is initialized with a temperature transition so that the density maximum occurs in a thin region near mid-tank. A shear layer with flow in the x, or along tank, direction is collocated with the temperature transition, and instability is triggered by white noise in the velocity field.

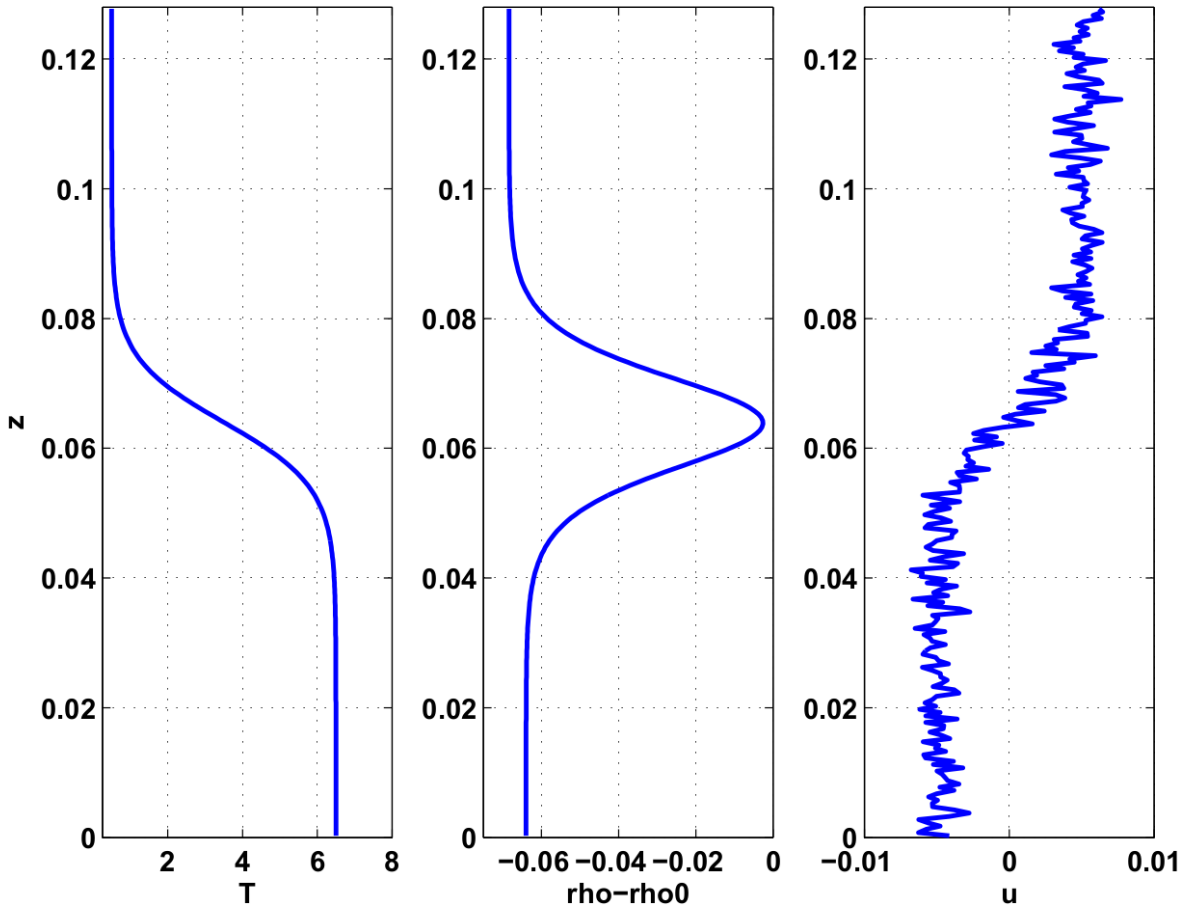

Figure 6.1: The initial state of the simulation. From left to right, the temperature, density perturbation, and velocity perturbation profiles.

Figure 6.1 shows the initial state of the simulation. The mid-depth density maximum combined with the noisy shear velocity profile induce competing Kelvin-Helmholtz and Rayleigh-Taylor instabilities.

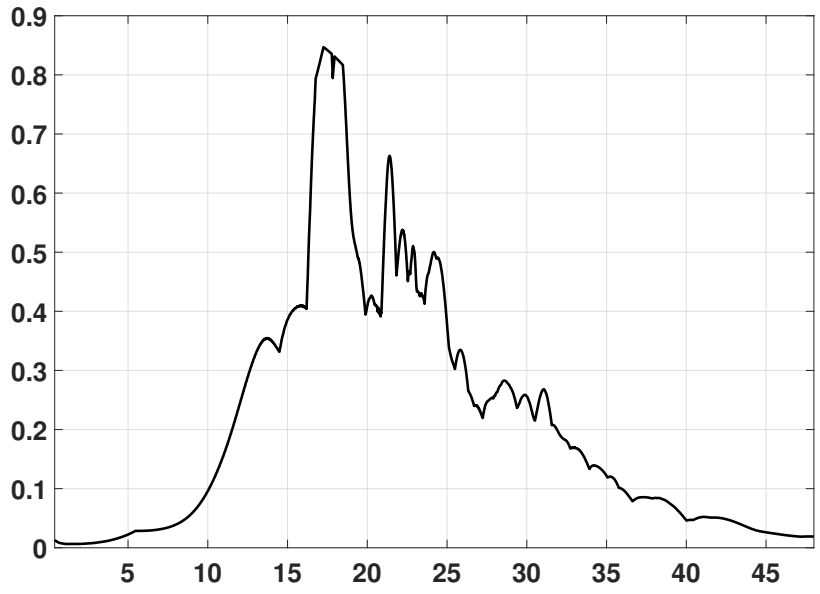

Figure 6.2: The gamma method results using kinetic energy, enstrophy, maximum viscous dissipation, and maximum vertical velocity for the defining set. Rather than choose a feature length we simply used the maxima in this case.

Figure 6.2 shows the results of the gamma method applied to the data set using the defining set of kinetic energy, enstrophy, maximum viscous dissipation, and maximum vertical velocity. In this case we do not have a clear choice for feature length, and so following the advice of section 2.3.4, we will simply investigate particular maxima of the gamma curve. Clearly outputs 17, 18, and 21 are good choices to consider.

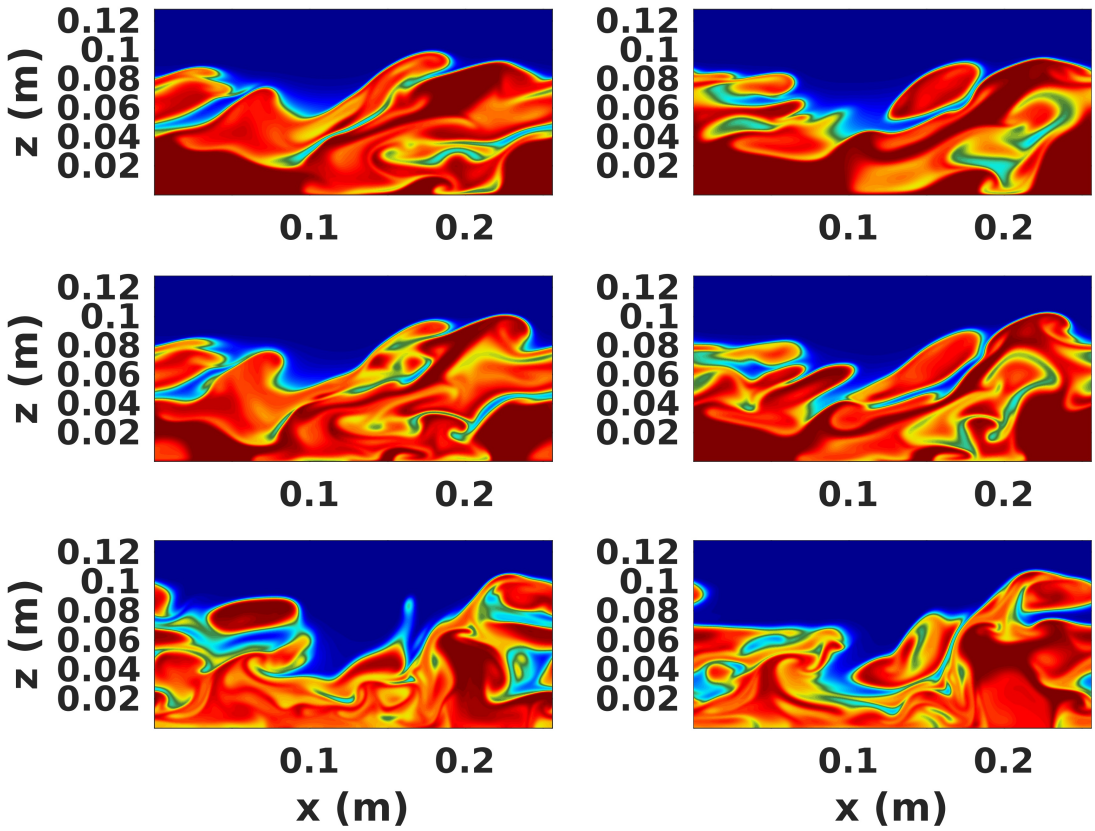

Figure 6.3: Outputs 17, 18, 21 of the temperature field from top to bottom, as chosen by the gamma method depicted in Figure 6.2. On the left we have slices at $y = 10$, and on the right slices at $y = 30$.

Figure 6.3 shows the outputs chosen by the gamma method at two different y locations. The gamma method has identified times where the two instabilities are causing mixing, but the structures are still large scale.

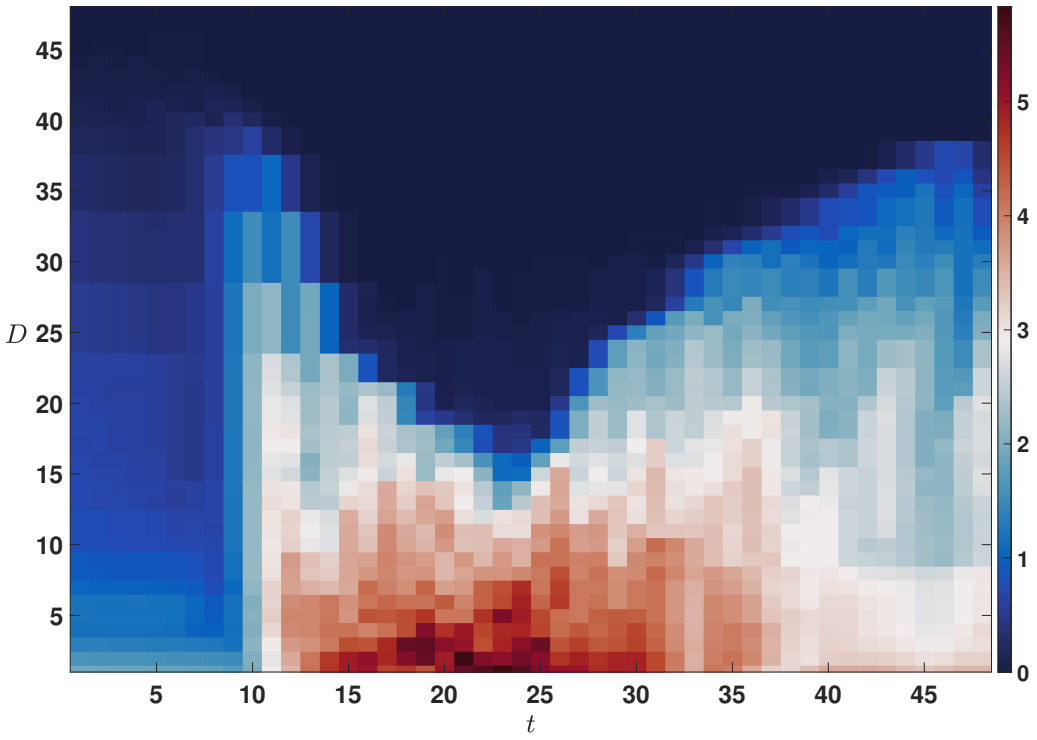

Figure 6.4: The error map for temperature field of the cabbeling data set.

Figure 6.4 shows the error map for the temperature field of the data set. Outputs 12, 23, and 47 were chosen as indicators of early, mid, and long time behaviour.

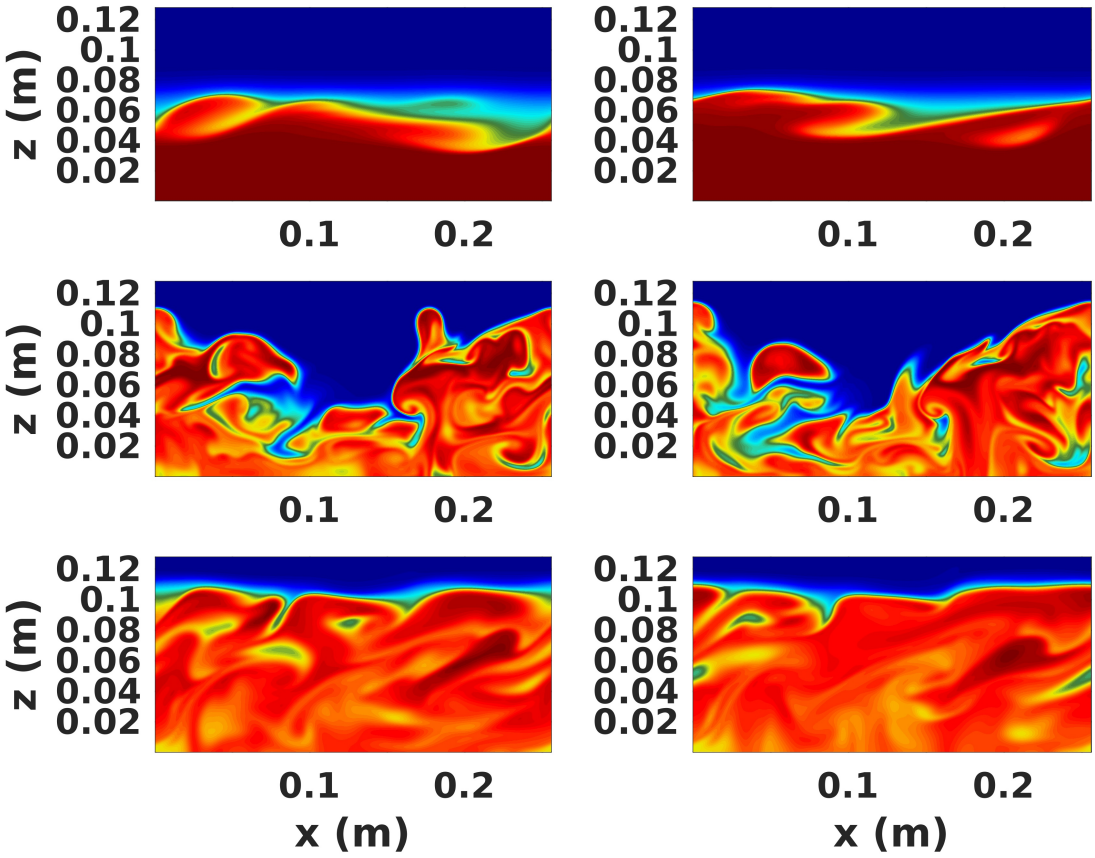

Figure 6.5: Outputs 12, 23, 47 of the temperature field from top to bottom, as indicated by the error map in Figure 6.4. On the left we have slices at $y = 10$, and on the right slices at $y = 30$.

Figure 6.5 shows slices at $y = 10$ and $y = 30$ for some times selected by looking at the error map. The large error structure around times 10 to 12 indicates the onset of the Kelvin-Helmholtz instability, while the end time behaviour is that of a Rayleigh-Taylor instability. These are indicated in the top and bottom rows of the Figure respectively. In between, around output 23 in the middle row of the Figure, both are occurring. We see that the error map clearly indicates these times of interest.

Note that the gamma results of 17, 18 and 21 are clustered within the major feature in

the error map centred at time 23. We see that there is good agreement between the two methods.

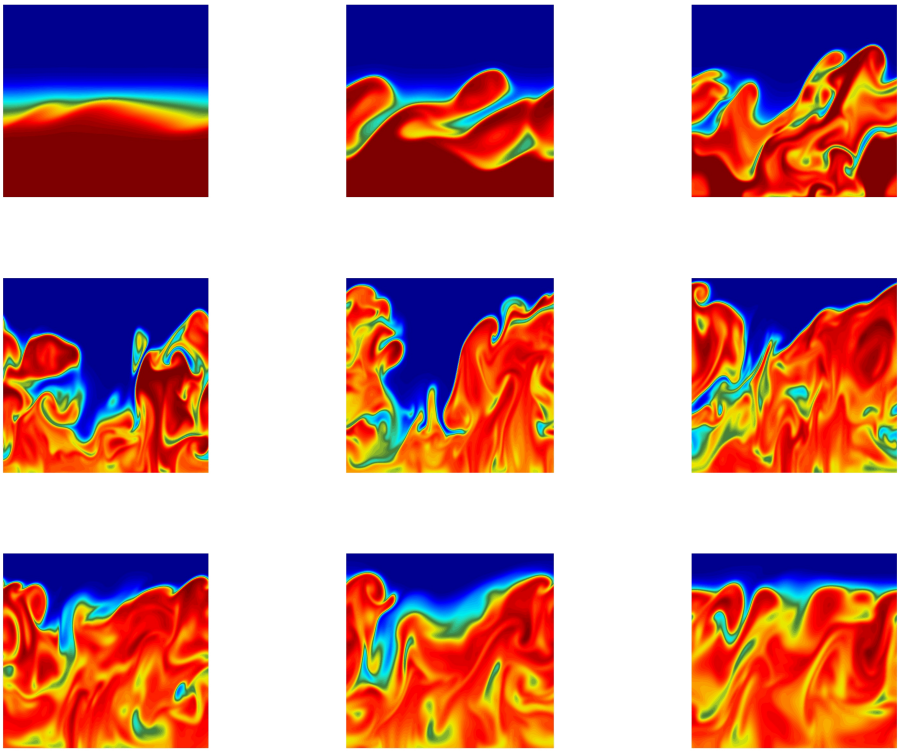

Figure 6.6: Time outputs 10, 14, 18, 22, 26, 30, 34, 38, 42 in order left to right and top to bottom. This is the temperature field with the most dense water at the mid depth.

Finally, as confirmation of our work, we show the evolution of the x-z temperature field at $y = 10$, as depicted in Figure 6.6. This confirms our intuition of what the error map was outlining: the development of Kelvin-Helmholtz instability followed by a period of intense mixing and cabbeling, which in turn drives further mixing. A "standard" shear instability lifecycle would not have the cabbeling driven production of dense water and hence the instability would "mix out" much earlier. The error map's butterfly shape

shows that the fully three-dimensionalized, cabbeling driven portion of the instability evolution produces a substantial amount of small scale features, which require a larger number of EOF modes to represent accurately.

6.2.2 Internal Seiche with Multiple Instability Types

This experiment is another of the runs in the parameter space exploration of the situation depicted in Figure 3.5 of section 3.1.2. The model tank is $(1.024, 0.128)$ m with a grid of $(6144, 768)$, for a resolution of 0.167 mm^2. There are 300 outputs 1 s apart. The early portion of the evolution consists of an internal seiche, and hence the error map is only computed for the second half of the simulation, or outputs 150-300.

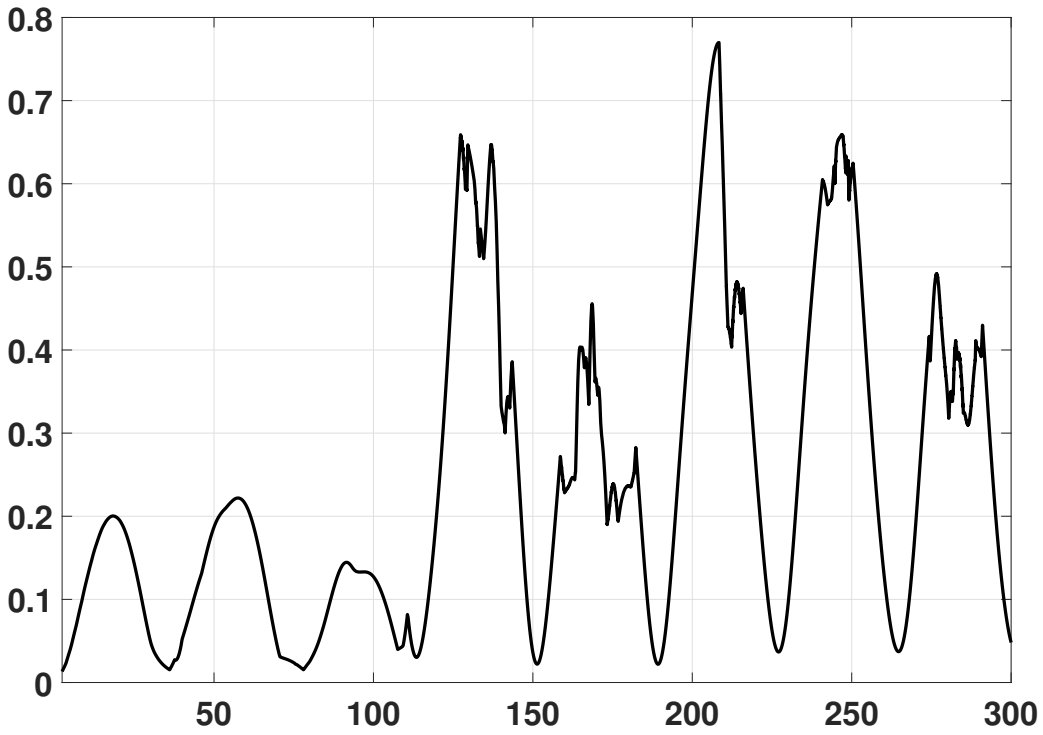

Figure 6.7: The Gamma Method on kinetic energy, enstrophy, max dissipation, and max vertical velocity. Maxima 127, 208, 247 were chosen.

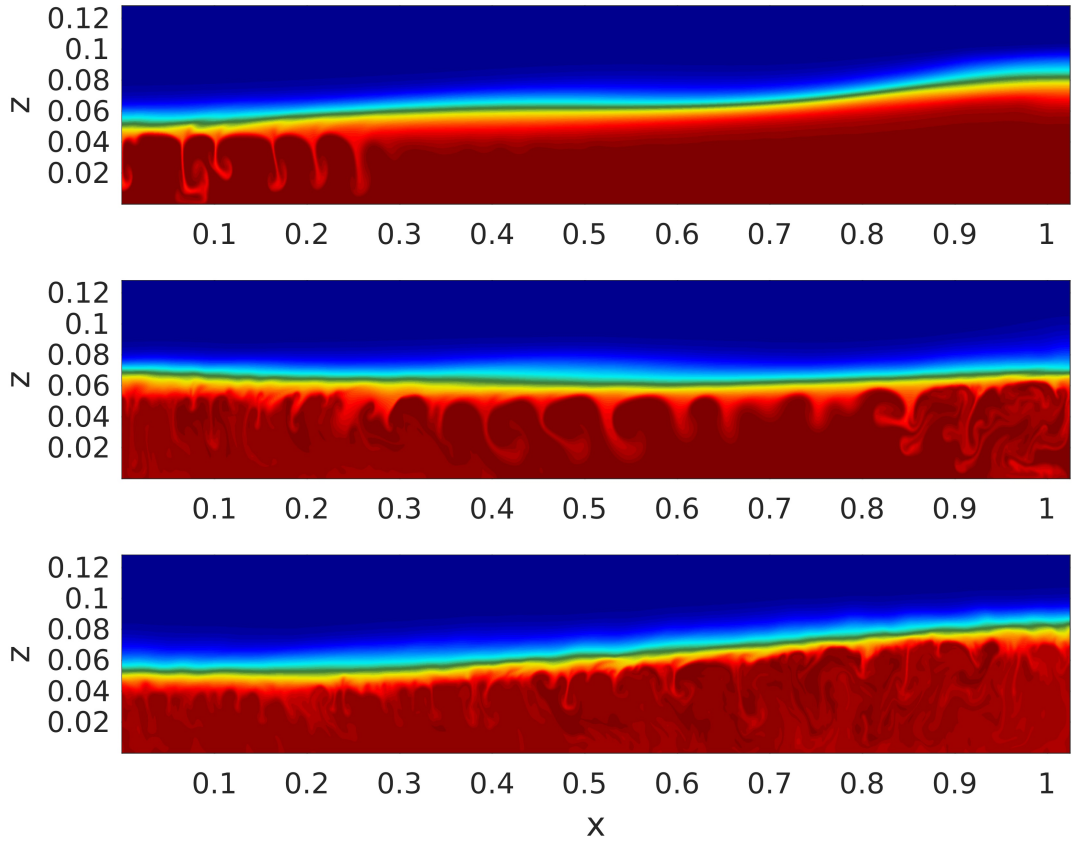

Figure 6.8: The gamma method showed maxima at 127, 208, and 247, and the temperature fields of these outputs are arranged from top to bottom.

Figure 6.7 shows the gamma method for this data set, and Figure 6.8 shows the outputs chosen by the gamma method. The top panel (output 127) depicts the onset of the double diffusive instability. The middle panel (output 208) depicts a time of propagating double diffusive instabilities. The bottom panel (output 247) shows the breakdown into small scale structures.

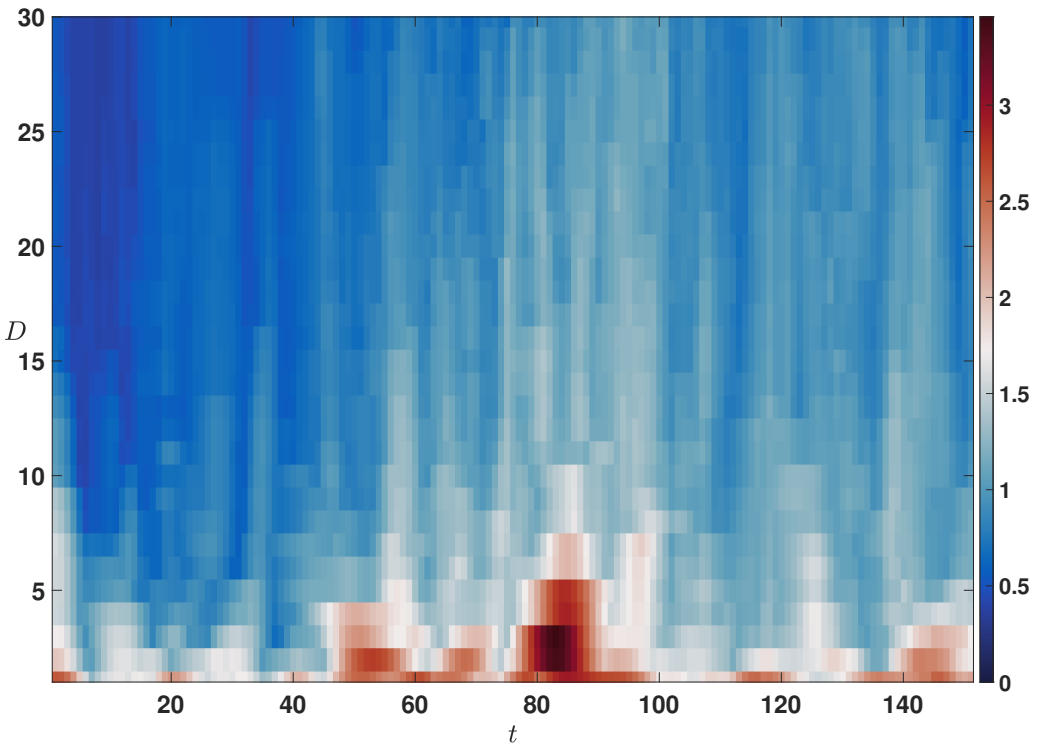

Figure 6.9: The error map on the salinity field for the last 150 outputs of the simulation, corresponding to the last half of the outputs in Figure 6.7. Only 30 modes were used, to reduce total computation time.

Figure 6.9 shows the error map for this data set. It was performed only on the final 150 outputs, as we expected more dynamics in the second half of the simulation. Moreover we chose to include only 30 modes to reduce computation time. These are the kind of choices which can be made in a particular context. We see several times of interest. We chose outputs 58, 82, and 145 in the last 150 outputs, corresponding to outputs 208, 232, and 295.

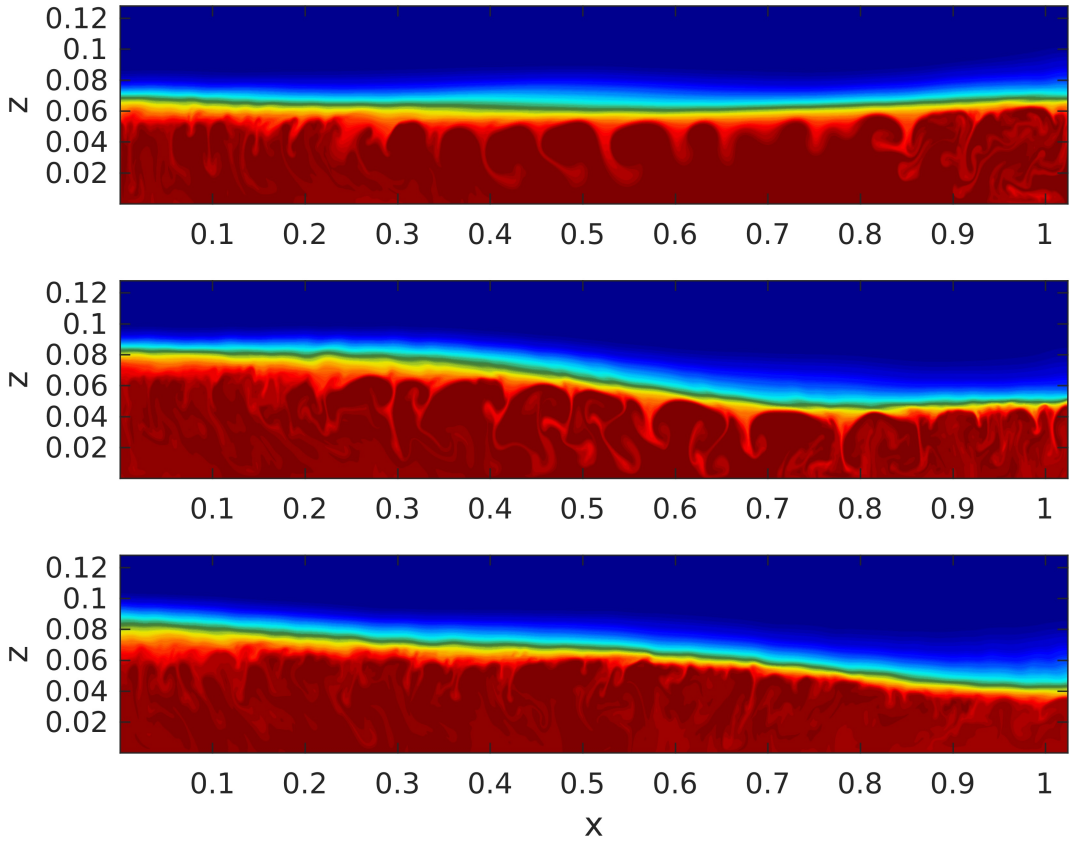

Figure 6.10: Temperature fields of outputs 208, 232, 295 as selected by the error map method. See text for details.

Figure 6.10 shows the outputs chosen using the error map of Figure 6.9. The top panel shows output 208, which was the second feature identified by the Gamma method of Figure 6.7, and depicted in the middle panel of Figure 6.8. This is the propagating double diffusive instability. The middle and bottom panels show the subsequent breakdown of this instability into finer structure.

It is interesting to note that the propagating, coherent, doubly diffusive instability of output 208 was chosen by both methods. A doubly confirmed time such as this is of primary interest.

6.3 Summary

Both the Gamma and EOF error map methods allow us to identify features. While the Gamma method runs in seconds, the error map in these cases took less than 12 hours. Future work would include improvements in this area, although the simplest fix is to move to a solid state hard drive to speed up multiplication. Currently 12 hours is sufficient for our needs, as the gamma method results are available immediately, and the error map results can be ready the following day.

Chapter 7

Conclusion

We began with the problem of finding interesting times in data sets. This problem was originally posed in the context of time series data sets collected from an *in situ* instrument cluster. A simultaneous perturbation argument led to the gamma method, which has proven to be effective on every time series data set to which it has been applied. We then considered the problem of finding interesting times in time-indexed model output. In this case we found that the gamma method was not suitable for application directly to the data set, but could still be applied to good effect using a time series data set constructed from the model output. This requires some expertise, but no more than the gamma method already requires of the intended user.

The EOF error map method was developed in order to take full advantage of the spatial information present in time-indexed model output. This method has proven to be effective in finding time periods of interest in every time-indexed model output data set to which it has been applied. Some especially large data sets required additional coding and runtime to get around resident memory limitations, but even in these cases the implementation is straightforward using built in MATLAB toolboxes.

Emboldened by our continual success on data sets with a time index, we next considered the problem of applying both methods to data sets without a time index. We concluded that the flexible nature of the gamma method would almost certainly have a valid application in static data sets, but that it would be too context dependent to say anything general here. In the case of the EOF error map method, we derived the ordered EOF error maps as a way of measuring how ensemble members represented the whole ensemble.

Our static data set was single time outputs of a set of dynamic data sets. We developed the first eigenvalue series for identifying time periods of interest in ensembles of time-indexed model output. As ensembles of dynamic data sets are not the norm, we have less experience in the application of this method. However the initial results are promising, and we view this as a prime candidate for future work.

We have made every attempt to relate every aspect of this thesis to the original problem of finding interesting times in data sets. As we have just discussed, this led to methods which solved this problem in time series data sets, time-indexed model output, and ensembles of time-indexed model output. Moreover we developed an analogous method for static data sets. We have thoroughly answered the original question.

References

[1] Hervé Abdi and Lynne J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.

[2] Athanasios C. Antoulas. *Approximation of Large-Scale Dynamical Systems*. Siam, Philadelphia, 2005.

[3] Tom Beer. *Environmental Oceanography: Second Edition*. Taylor & Francis Group, Boca Raton, 1997.

[4] Gaëlle Casagrande, Yann Stephan, Alex C. Warn Varnas, and Thomas Folegot. A Novel Empirical Orthogonal Function (EOF)-Based Methodology to Study the Internal Wave Effects on Acoustic Propagation. *IEEE Journal of Oceanic Engineering*, 36(4):745–759, 2011.

[5] Ishanu Chattopadhyay and Hod Lipson. Data smashing: uncovering lurking order in data. *Journal of the Royal Society Interface*, 11:20140826, 2014.

[6] Frédéric Chazal, Marc Glisse, and Bertrand Michel. Convergence Rates for Persistence Diagram Estimation in Topological Data Analysis. *Journal of Machine Learning Research*, 16:3603–3635, 2015.

[7] Aaron Coutino, Marek Stastna, Shawn Kovacs, and Eduard Reinhardt. Hurricanes Ingrid and Manuel (2013) and their impact on the salinity of the Meteoric Water Mass, Quintana Roo, Mexico. *Journal of Hydrology*, 551:715–729, 2017.

[8] P. A. Davidson. *Turbulence: An Introduction For Scientists and Engineers*. Oxford Univ Press, 2015.

[9] David Deepwell, Marek Stastna, and Aaron Coutino. Multi-scale phenomena of rotation-modified mode-2 internal waves. *Nonlinear Processes in Geophysics*, 25(1):217–231, 2018.

[10] William J Emery and Richard E Thomson. *Data Analysis Methods in Physical Oceanography*. Pergamon Press Ltd, 1998.

[11] Gary Froyland and Kathrin Padberg-Gehle. A rough-and-ready cluster-based approach for extracting finite-time coherent sets from sparse and incomplete trajectory data. *Chaos*, 25(8):087406, aug 2015.

[12] Dorian Fructus, Magda Carr, John Grue, Atle Jensen, and Peter A. Davies. Shear-induced breaking of large internal solitary waves. *Journal of Fluid Mechanics*, 620:1–29, 2009.

[13] Matan Gavish and David L. Donoho. The optimal hard threshold for singular values is $4/\sqrt{3}$. *IEEE Transactions on Information Theory*, 60(8):5040–5053, 2014.

[14] Alireza Hadjighasem, Mohammad Farazmand, Daniel Blazevski, Gary Froyland, and George Haller. A critical comparison of Lagrangian methods for coherent structure detection. *Chaos*, 27(5), 2017.

[15] Alireza Hadjighasem, Daniel Karrasch, Hiroshi Teramoto, and George Haller. Spectral-clustering approach to Lagrangian vortex detection. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 93(6), 2016.

[16] Azzam Haidar, Khairul Kabir, Diana Fayad, Stanimire Tomov, and Jack Dongarra. Out of memory SVD solver for big data. *2017 IEEE High Performance Extreme Computing Conference, HPEC 2017*, (Icl), 2017.

[17] A Hannachi, IT Jolliffe, and DB Stephenson. Empirical orthogonal functions and related techniques in atmospheric science: A review. *International Journal of Climatology*, 27(May):1119–1152, 2007.

[18] Richard J Hathaway and James C Bezdek. Extending fuzzy and probabilistic clustering to very large data sets. *Computational Statistics and Data Analysis*, 51(1):215–234, 2006.

[19] David J Hill and Barbara S Minsker. Anomaly detection in streaming environmental sensor data: A data-driven modeling approach. *Environmental Modelling and Software*, 25(9):1014–1022, 2010.

[20] Phillip Holmes, John Lumley, Gahl Berkooz, and Clarence Rowley. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, Cambridge, second edi edition, 2012.

[21] P.J. Huber. Massive Data Sets Workshop: The Morning After. In *Massive Data Sets. Proceedings of a Workshop*, Washington DC, 1996. National Academy Press.

[22] P.J. Huber and E.M. Ronchetti. *Robust Statistics: Second Edition*. John Wiley & Sons, 2009.

[23] James W Hurrell and Clara Deser. North Atlantic climate variability: The role of the North Atlantic Oscillation. *Journal of Marine Systems*, 79(3-4):231–244, 2010.

[24] Ilse C. F. Ipsen. *Numerical Matrix Analysis*. SIAM, 2009.

[25] Charles M. Judd, Gary H. McClelland, and Carey S. Ryan. *Data Analysis: A Model Comparison Approach*. Routledge, aug 2008.

[26] James M Kaihatu, Robert A Handler, George O Marmorino, and Lynn K Shay. Empirical orthogonal function analysis of ocean surface currents using complex and real-vector methods. *Journal of Atmospheric and Oceanic Technology*, 15(4):927–941, 1998.

[27] Chris Karhl, Keith Law, Jeff Bower, Jeff Hildebrand, Rany Jazayerli, Dave Pease, Steven Rubio, Joseph S Sheehan, Greg Spira, Michael Wolverton, Keith Woolner, and Clay Davenport. *Baseball Prospectus 2000*. Potomac Books Inc, 2000.

[28] Gaetan Kerschen, Jean Claude Golinval, Alexander F. Vakakis, and Lawrence A. Bergman. The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview. *Nonlinear Dynamics*, 41(1-3):147–169, aug 2005.

[29] Kwang Y. Kim and Qigang Wu. A comparison study of EOF techniques: Analysis of nonstationary data with periodic statistics. *Journal of Climate*, 12(1):185–199, jan 1999.

[30] Václav Kolář. Vortex identification: New requirements and limitations. *International Journal of Heat and Fluid Flow*, 28(4):638–652, 2007.

[31] Pijush K. Kundu, Ira M. Cohen, and David R. Dowling. *Fluid Mechanics: Fifth Edition*. Academic Press, Oxford, 2012.

[32] Vladimir Kurbalija, Miloš Radovanović, Zoltan Geler, and Mirjana Ivanović. A Framework for Time-Series Analysis. In *Artificial Intelligence: Methodology, Systems, and Applications*, volume 6304, pages 42–51. 2010.

[33] J. Nathan Kutz. *Data-Driven Modeling & Scientific Computation: Methods for Complex Systems and Big Data*. Oxford University Press, 2013.

[34] Kevin G Lamb. A numerical investigation of solitary internal waves with trapped cores formed via shoaling. *Journal of Fluid Mechanics*, 451:109–144, 2002.

[35] Kevin G. Lamb and David Farmer. Instabilities in an Internal Solitary-like Wave on the Oregon Shelf. *Journal of Physical Oceanography*, 41(1):67–87, 2011.

[36] Alexey Lyubushin. Global coherence of GPS-measured high-frequency surface tremor motions. *GPS Solutions*, 22(4):116, 2018.

[37] Alexey Lyubushin. Synchronization of Geophysical Field Fluctuations. In *Complexity of Seismic Time Series*, pages 161–197. 2018.

[38] Christopher V. Maio, Jeffrey P. Donnelly, Richard Sullivan, Stephanie M. Madsen, Christopher R. Weidman, Allen M. Gontz, and Vitalii A. Sheremet. Sediment dynamics and hydrographic conditions during storm passage, Waquoit Bay, Massachusetts. *Marine Geology*, 381(October 2017):67–86, 2016.

[39] David S Matteson and Nicholas A James. A nonparametric approach for multiple change point analysis of multivariate data. *Journal of American Statistical Association*, 109(505):334–345, 2014.

[40] Igor Mezić. Analysis of Fluid Flows via Spectral Properties of the Koopman Operator. *Annual Review of Fluid Mechanics*, 45(1):357–378, 2013.

[41] M Mourad and J.-L. Bertran-Krajewski. A method for automatic validation of long time series of data in urban hydrology. *Water Science and Technology*, 45(4-5):263–270, 2002.

[42] Karl Pearson. Principal Components Analysis. *The London, Edinburgh and Dublin Philosophical Magazine and Journal*, 6(2):566, 1901.

[43] Daniel Pena and Pilar Poncela. Dimension Reduction in Multivariate Time Series. In *Advances on Distribution Theory, Order Statistics and Inference, in Honor of B. C. Arnold*, number 1981, pages 836–843. 2006.

[44] Timour Radko. Thermohaline layering in dynamically and diffusively stable shear flows. *Journal of Fluid Mechanics*, 805:147–170, 2016.

[45] Kexin Rong and Peter Bailis. ASAP: Prioritizing Attention via Time Series Smoothing. *Proceedings of the VLDB Endowment*, 10(11):1358–1369, 2017.

[46] Clarence W Rowley, Tim Colonius, and Richard M Murray. Model reduction for compressible flows using POD and Galerkin projection. *Physica D*, 189:115–129, 2004.

[47] Clarence W Rowley and Scott T.M. Dawson. Model Reduction for Flow Analysis and Control. *Annual Review of Fluid Mechanics*, 49(1):387–417, 2017.

[48] Kristy L. Schlueter-Kuck and John O. Dabiri. Identification of individual coherent sets associated with flow trajectories using coherent structure coloring. *Chaos*, 27(9), 2017.

[49] Peter J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656(4):5–28, 2010.

[50] Onofrio Semeraro, Gabriele Bellani, and Fredrik Lundell. Analysis of time-resolved PIV measurements of a confined turbulent jet using POD and Koopman modes. *Experiments in Fluids*, 53(5):1203–1220, 2012.

[51] Justin Shaw and Marek Stastna. Feature identification in time-indexed model output. *PLoS ONE*, 14(12):1–17, 2019.

[52] Justin Shaw, Marek Stastna, Aaron Coutino, Ryan K. Walter, and Eduard Reinhardt. Feature identification in time series data sets. *Heliyon*, 5, 2019.

[53] N. C. Shibley and M. L. Timmermans. The Formation of Double-Diffusive Layers in a Weakly Turbulent Environment. *Journal of Geophysical Research: Oceans*, 124(3):1445–1458, 2019.

[54] Nancy Soontiens, Marek Stastna, and Michael L Waite. Trapped internal waves over topography: Non-Boussinesq effects, symmetry breaking and downstream recovery jumps. *Physics of Fluids*, 25(6), 2013.

[55] Marek Stastna, Jason Olsthoorn, Anton Baglaenko, and Aaron Coutino. Strong mode-mode interactions in internal solitary-like waves. *Physics of Fluids*, 27(4):46604, 2015.

[56] Christopher Subich, Kevin G Lamb, and Stast. Simulation of the Navier–Stokes equations in three dimensions with a spectral collocation method. *International Journal for Numerical Methods in Fluids*, 73:103–129, 2013.

[57] M Sudharsan, Steven L Brunton, and James J Riley. Lagrangian coherent structures and inertial particle dynamics. *Physical Review E*, 93(3), 2016.

[58] Mohamed H.M. Sulman, Helga S. Huntley, B. L. Lipphardt, and A. D. Kirwan. Leaving flatland: Diagnostics for Lagrangian coherent structures in three-dimensional flows. *Physica D: Nonlinear Phenomena*, 258:77–92, 2013.

[59] David W. J. Thompson and John M. Wallace. The Arctic oscillation signature in the wintertime geopotential height and temperature fields. *Geophysical Research Letters*, 25(9):1297–1300, 1998.

[60] Kristen M Thyng, Chad A Greene, Robert D Hetland, Heather M Zimmerle, and Steven F Dimarco. True Colors of Oceanography Guidelines for Effective and Accurate Colormap Selection. *Oceanography*, 29(3):9–13, 2016.

[61] Christopher Torrence and Gilbert P. Compo. A Practical Guide to Wavelet Analysis. *Bulletin of the American Meteorological Society*, 79(1):61–78, jan 1998.

[62] Lloyd N. Trefethen and David. Bau. *Numerical linear algebra*, volume 50. Siam, 1997.

[63] A T Walden and A Serroukh. Wavelet analysis of matrix-valued time-series. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 458(2017):157–179, 2002.

[64] Ryan K. Walter, Emma C. Reid, Kristen A. Davis, Kevin J. Armenta, Kevin Merhoff, and Nicholas J Nidzieko. Local diurnal wind-driven variability and upwelling in a small coastal embayment. *Journal of Geophysical Research: Oceans*, 122(2):955–972, 2017.

[65] Ryan K Walter, Marek Stastna, C Brock Woodson, and Stephen G Monismith. Observations of nonlinear internal waves at a persistent coastal upwelling front. *Continental Shelf Research*, 117:100–117, 2016.

[66] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[67] Zhaohua Wu, Norden E Huang, Steven R Long, and C.-K. Peng. On the trend, detrending, and variability of nonlinear and nonstationary time series. *Proceedings of the National Academy of Sciences*, 104(38):14889–14894, 2007.

[68] Zhi Xiong, Qingrun Zhang, Alexander Platt, Wenyuan Liao, Xinghua Shi, Gustavo de los Campos, and Quan Long. OCMA: Fast, Memory-Efficient Factorization of Prohibitively Large Relationship Matrices. *G3 Genes—Genomes—Genetics*, 9(1):13–19, 2019.

[69] Chengzhu Xu, Marek Stastna, and David Deepwell. Spontaneous instability in internal solitary-like waves. *Physical Review Fluids*, 4(1):14805, 2019.

[70] Kiyoung Yang and Cyrus Shahabi. A PCA-based similarity measure for multivariate time series. In *Proceedings of the 2nd ACM international workshop on Multimedia databases - MMDB '04*, page 65, 2004.

[71] Qingshan Zhang, Yingzheng Liu, and Shaofei Wang. The identification of coherent structures using proper orthogonal decomposition and dynamic mode decomposition. *Journal of Fluids and Structures*, 49:53–72, 2014.

[72] Yuan Zhang, John M Wallace, and David S Battisti. ENSO-like interdecadal variability: 1900-93. *Journal of Climate*, 10(5):1004–1020, 1997.

Appendix A

Appendix

A.1 Gamma on CFD data: Zero Contours

In section 3.2 we saw that the gamma method was not well suited to finding features in time-indexed model output in a way that considered spatial information. The best option available for the gamma method in these contexts is to construct a time series data set from the model output and apply gamma to the time series data sets as it was designed to do. Essentially a strategy of ignoring spatial information to find features in time. Another strategy is to ignore the time information to find features in space. To see how this would work, again consider the dual pycnocline data set introduced in section 3.1, and in Figure 3.9. This data set actually contains not just density, but horizontal velocity u and vertical velocity w. The density perturbations of Figure 3.9, along with the velocity perturbation fields are shown in the top three panels of Figures A.1, A.2, A.3, and A.4, mirroring the time outputs in Figure 3.9.

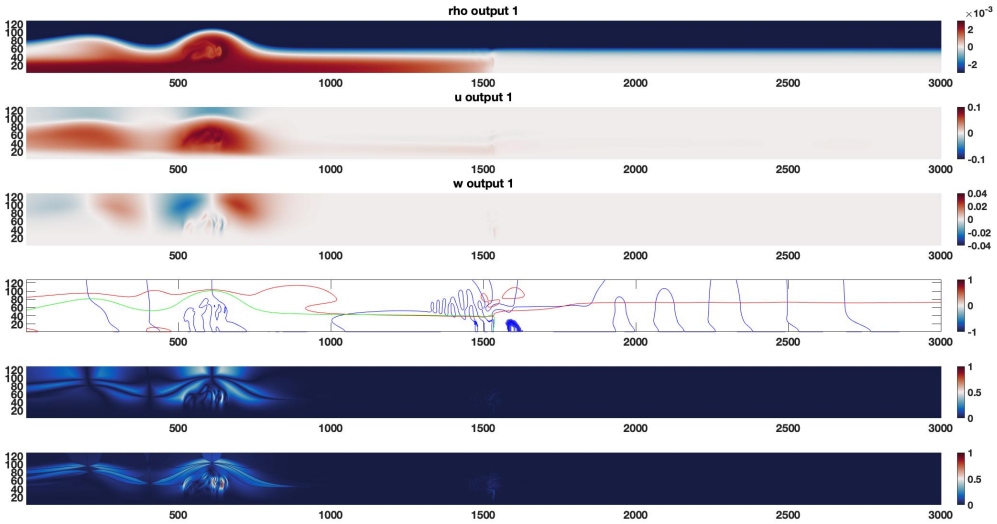

Figure A.1: The zero contour gamma method applied to the Dual Pycnocline case. Time output 1, corresponding to the top panel of Figure 3.9. From top to bottom the panels are density ρ, horizontal velocity u, vertical velocity w, zero contours for all three data sets (ρ in green, u in red, w in blue), the gamma field, and the visual gradient of the gamma field.

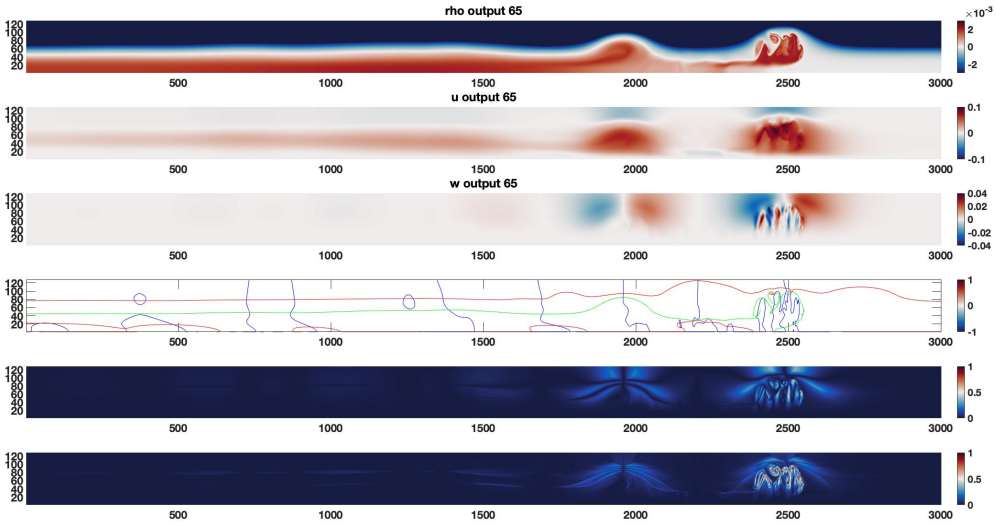

Figure A.2: The zero contour gamma method applied to the Dual Pycnocline case. Time output 65, corresponding to the second panel of Figure 3.9. From top to bottom the panels are density ρ, horizontal velocity u, vertical velocity w, zero contours for all three data sets (ρ in green, u in red, w in blue), the gamma field, and the visual gradient of the gamma field.

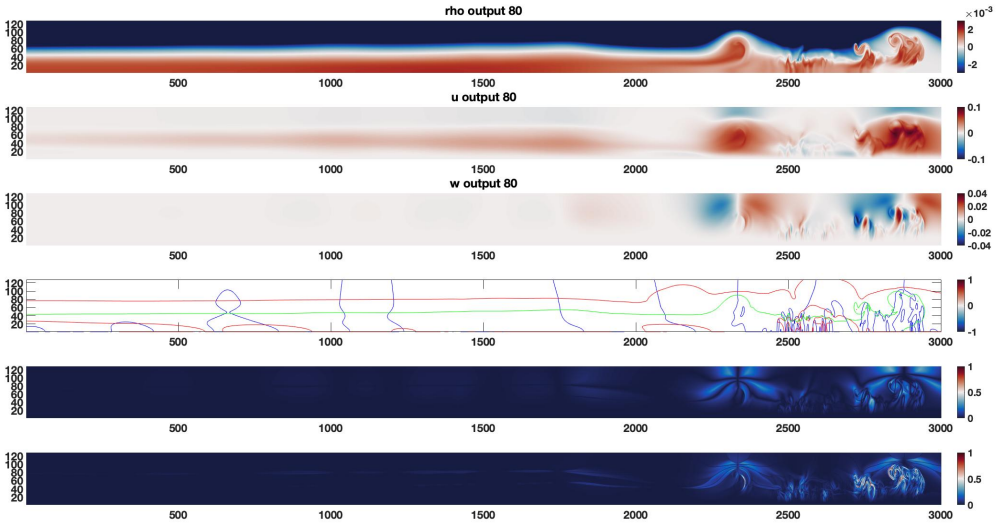

Figure A.3: The zero contour gamma method applied to the Dual Pycnocline case. Time output 80, corresponding to the third panel of Figure 3.9. From top to bottom the panels are density ρ, horizontal velocity u, vertical velocity w, zero contours for all three data sets (ρ in green, u in red, w in blue), the gamma field, and the visual gradient of the gamma field.

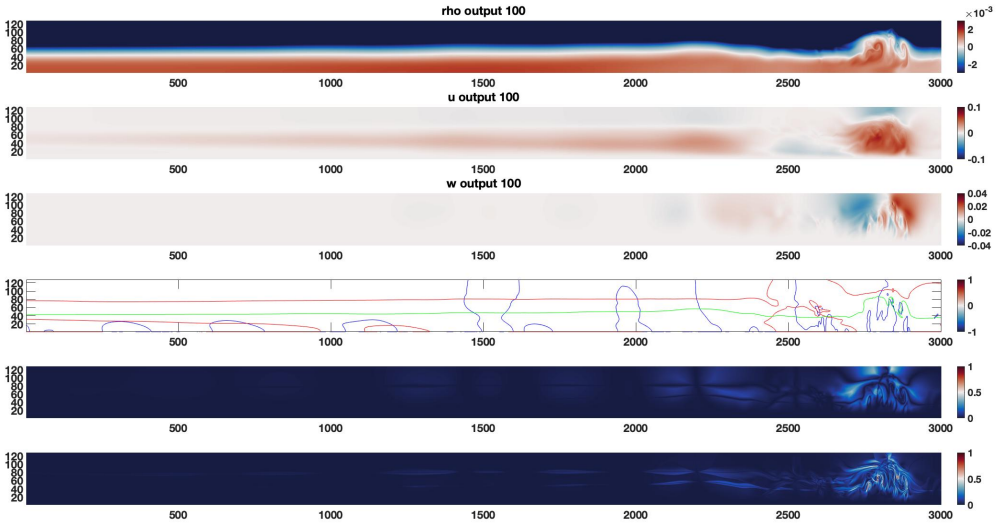

Figure A.4: The zero contour gamma method applied to the Dual Pycnocline case. Time output 100, corresponding to the bottom panel of Figure 3.9. From top to bottom the panels are density ρ, horizontal velocity u, vertical velocity w, zero contours for all three data sets (ρ in green, u in red, w in blue), the gamma field, and the visual gradient of the gamma field.

We are now looking for interesting locations within each time output. We apply the spirit of the gamma method, using our domain knowledge to guide our progress. We know that shear plays an important role in stratified flow dynamics. As these are perturbation fields, the zero contour for the density field marks the midpoint of the waveguide and the zero contours of the velocity fields mark locations where shear may develop. The zero contours are therefore related to locations of interest, because they identify either locations of propagation or instability development. These zero contours are plotted in the 4th panels of Figures A.1, A.2, A.3, and A.4.

Unfortunately this visualization is poor, as it draws the eye to zero contours of w which are not as important as the image makes it seem. This is because the zero contours are not sufficient conditions for dynamics, and so a zero contour in a nearly zero field is not interesting, but is highlighted anyway. However multiple zero contours nearby one another indicate regions of possible shear and propagation, and so should be considered. To find these we use all three fields in an analogue to the gamma method. We apply the

gamma method to the slice, using the three physical fields as the defining set with no de-trending as we ignore time information, and scaling by the maximum of each respective field. This produces what we'll call the gamma field at each time. It is simply the minimum of the absolute value of the three scaled physical perturbation fields in the data set at each time. The gamma field is presented in the 5th panels of each of Figures A.1, A.2, A.3, and A.4.

One final improvement on this visualization is to highlight the zero contour clusters. As an image, clusters of zero contours are characterized by rapid changes of color values in space. MATLAB's `imgradient` function from the image processing toolbox can be used to highlight such areas. The bottom panel in each figure is the magnitude of the image gradient of the gamma field. It clearly shows the small scale structure in the gamma field, as that is where there is a concentration of zero contours from one of the three physical fields. This omits dynamically uninteresting zero contours, while identifying spatial features. If you look closely you can also see the sharp change in density near the middle of the domain.

Clearly we had to go pretty far afield from our original method to find a way to apply the gamma method in a way that considered spatial information. So much so that we had to abandon the original problem of finding interesting times in the interest of finding interesting locations. The extension of the gamma method to zero contours within fluid dynamics data sets changes its function to a visualization tool within that specific context. It is a valid criticism of this extension to say that looking at each of the three fields also clearly identifies where the features are, and so there is little need for the gamma method. However it is also true that the gamma field gives a way to look at all three physical fields at once, through the lens of the identification of important dynamics. While it has this value, we still regard it as an aesthetically pleasing dead end.

A.2 Perception in the Analysis Pipeline

A few years ago, [60] introduced perceptually uniform colormaps to the oceanographic community, of which we are a part. As they put it "In a perceptually uniform colormap, any step in the map is perceived by the viewer to be the same size as any equally sized step elsewhere in the colormap." This prevents artificial gradients, and also ensures that every gradient is represented. These are color maps that neither preferentially exaggerate nor obscure the underlying data. Thyng et al's paper [60] includes several examples of

perceptually uniform colormaps from their `cmocean` package outperforming standard colormaps (e.g. MATLAB's jet colormap), in a variety of oceanographic contexts.

As impressive as they are, perceptually uniform color maps are not widely used. This led to a great deal of discussion on how perception relates to common practices. For instance, data sets are often compressed or filtered before they are visualized. How much of a perceptual difference do these processes make on the final visualization when compared to the original data? As we will see in section A.2.2, D-EOF reconstructions can be thought of as a compressed version of the original data set. As discussed in section 4.3.5 reconstructions can also be thought of as a variance filtered version of the original data. We decided to use EOF as an example of both compression and filtering methods, and set out to examine their induced perceptual change using D as a parameter.

We needed a way to quantify the perceptual differences between the visualizations of raw and processed data. We chose the Structural Similarity (SSIM) index of [66], which is able to quantify the perceptual differences between the grayscale visualization of the raw data and the corresponding processed data. This is accomplished by comparing the two images through the use of a structure function based on correlation. This structure function is supported by a variance normalization and a luminance centering operation [66]. Each component is based on a known aspect of the human visual system, which makes the SSIM a grayscale image quality metric which quantifies error based on how visible that error is to the human eye, rather than on an exclusively mathematical basis. Figure 2 of [66] (where SSIM was introduced) gives the example of several degraded images having nearly identical mean-square error (MSE) with the original image, but varying widely in perceptual quality. This example shows that standard mathematical metrics can fail to distinguish between perceptually different images. To quantify perceptual changes caused by filter and compression method selection, we employ the Structural Similarity (SSIM) index. The grayscale image of the raw data acts as the reference image, and grayscale images of the EOF reconstructions as reduced quality images of the reference. The method produces an SSIM map the same size as the images it is applied to, where the values in the map are 1 if there is perceptual local agreement, and is lower the less local perceptual agreement there is. In our implementation, the MATLAB function `mat2gray` was used to create grayscale images of the raw data and each reconstruction for SSIM analysis. To ensure the grayscale maps takes the same values across images, the minimum and maximum over all reconstructions and data values were taken as the bounds for `mat2gray`. The MATLAB `ssim` command was then used to obtain the SSIM maps, along with the mean SSIM (MSSIM) value.

It is unfortunate that SSIM only applies to grayscale images, but the SSIM is based on the assumption that the human visual system is highly adapted for extracting structural

information, and grayscale is sufficient for this purpose. Moreover this is understandable from a mathematical viewpoint, as every pixel in a grayscale image can be represented by a single real number, but every pixel in a colour image requires three real numbers. SSIM has been very successful, and so has the advantage of being well established and implemented in MATLAB. This severely reduces the barrier to entry for anyone who would wish to replicate our work.

A.2.1 The Monterey Bay Data Set

We used this data set as an example of the gamma method in section 2.4.1. In this section we will consider a subset of this data set, whose oceanographic features are described in detail in [65]. Briefly, the source of this data set is a combined moored array of instruments and an acoustic Doppler current profiler (ADCP) deployed in Monterey Bay, Cailfornia between the 7th and 22nd of July, 2011. The array was deployed where the California Current and its associated upwelling of cold water interacts with warm water found within Monterey Bay that is shielded form offshore winds. For this reason this region is referred to as the 'upwelling shadow.' The dynamics at this location yield episodes of highly energetic fronts that move past the instrument array, and subsequently break up into a combination of large amplitude internal waves and instabilities. These waves and instabilities also propagate past the measurement array. Walter et al. [65] carry out their analysis at two timescales. The longer term record is split into two time periods with wave activity and one in which the larger scale upwelling precludes wave activity. On a shorter timescale, they detail a frontal crossing on 17 July, 2011 (their Figure 5) in which large-amplitude internal waves in the presence of background shear were too large to be described by all existing internal wave theories. This was unique in the literature up to that time. Since the coastal environment in Monterey Bay is believed to be representative of other geographical locations (e.g. the Peru-Chile current system) and other eastern boundary current upwelling systems around the world, it is important to quantitatively characterize the observed features. This is especially true since existing theories of internal waves have proven inadequate.

Since the focus of this section is on methodology, we will consider only the detailed measurements of the normalized kinetic energy ($\frac{1}{2}\left(u^2+v^2+w^2\right)$) during the previously detailed frontal crossing of 17 July, 2011. This corresponds to the gamma method $\mathcal{F}_1$ of Figure 2.2 panel c. While the front is associated with water that is up to 5 degrees warmer than that found offshore, all the features of motion necessary for analysis can be identified in the kinetic energy field. It is this field and time period which will serve as

the data set example throughout this section.

The dimensions of the data set during this period are 35×1301. Because this section is about images, the axis labels in all of the image Figures (A.5, A.6, A.7, and A.8) are simply pixel number. We refer the reader to [65] for a discussion of the physical dimensions of this data set.

A.2.2 Compression by EOF

The size of the raw data is just the dimension of the associated data matrix $\mathbf{X}$, namely $MN = 35 \times 1301 \approx 4.5 \times 10^4$ for the Monterey Bay data set. The size of the reconstruction using D modes is

$$\begin{aligned} &(\text{size of } D \text{ coefficients}) + (\text{size of } D \text{ EOFs}) && \text{(A.1)} \\ &= DN + DM && \text{(A.2)} \\ &= D(N + M) && \text{(A.3)} \end{aligned}$$

Setting $D = M$ shows that keeping all EOF information results in M^2 more data than the original signal. For compression

$$D < \frac{MN}{N + M} = \frac{M}{1 + \frac{M}{N}} \tag{A.4}$$

In the Monterey Bay data set $M = 35 \ll 1301 = N$, so that $\frac{M}{N} \ll 1$. It is common for geophysically sourced data sets to have $M \ll N$ (i.e. many more points in time than in space). In this case if $D \leq M - 1$ the EOF reconstruction compresses the raw data. We will see in section A.2.4 that in general there is no need to take this many modes, so that an EOF reconstruction nearly always compresses the raw geophysically sourced data sets. The amount of compression is given by

$$\frac{D(M + N)}{MN} = \frac{D}{\frac{M}{1+\frac{M}{N}}} \approx \frac{D}{M} \tag{A.5}$$

So when the raw data has $M \ll N$, the EOF reconstruction with D modes is approximately D/M % of the original data's size. By symmetry when $N \ll M$ the approximation is D/N %. Either way we see that EOF reconstructions can be thought of as compressed versions of the original data set.

A.2.3 Singular Value Hard Thresholding

The SSIM index quantifies perceptual error. Since EOF reconstructions converge to the data, how large does D have to be before the analyst cannot tell the difference between the visualizations of the D EOF representation and the data? The EOF basis is constructed so that each EOF added to the representation adds less variance than the previous one. We would expect that at some point the added EOF makes very little difference to the perception of the resulting visualization of that truncated EOF representation. That is, we expect that there is a $D < M$ for which there is very little perceptual error. Writing equation 4.17 is roughly equivalent to stating that the important part of the data is a matrix whose rank is (much) lower than the size of the raw data matrix. The raw data matrix may then be thought of as the 'true' data with some added noise. For a given data set, it may be that both the rank of the matrix to be recovered, and the nature of the noise, are unknown. In a certain asymptotic framework, and assuming white noise, [13] outline a choice of D which is asymptotic MSE optimal, without knowledge of the rank. Their formulation is from the SVD perspective, and they call the truncated version of equation 4.10 'Singular Value Hard Thresholding' (SVHT) because it corresponds to setting all singular values greater than some cutoff called the 'hard threshold' to zero. Their method for choosing this hard threshold is called 'Optimal SVHT.' We used their code to choose D by optimal SVHT throughout this work. Note their code requires $M \leq N$ for the raw data matrix. It is often the case for geophysically sourced data, as it is for the Monterey Bay data set described in section A.2.1, that $M \ll N$. The reader is referred to [13] for the details on this method.

As we discussed in section 4.3.5, while mathematical choices must be made in order to make mathematical progress, it is the repeated success of the mathematical results in a given application which makes them of practical worth. In our examples, optimal SVHT produces a reconstruction whose visualization is nearly indistinguishable from that of the raw data. In this way optimal SVHT answers the question at the beginning of the previous paragraph, as we will show in section A.2.4. Pursuant to our discussion in section 4.3.5, this is a case where a static choice for D can be made mathematically, and where the underlying choice of mathematical framework is justified heuristically by continued success in the applications. As a rough guide [13] table IV and equation 4 shows that the hard threshold when $M \ll N$ is approximately 1.5 times the median unnormalized EOF singular value. While this choice for D is an approximation of a result based on their asymptotic framework, it is also clear that such a result may have developed heuristically in a given application. In either case the justification is continued success.

A.2.4 Results

Let us consider some examples of the perceptual difference between this Monterey bay data set and EOF-filtered or compressed versions of it. We consider a few different values of D as examples.

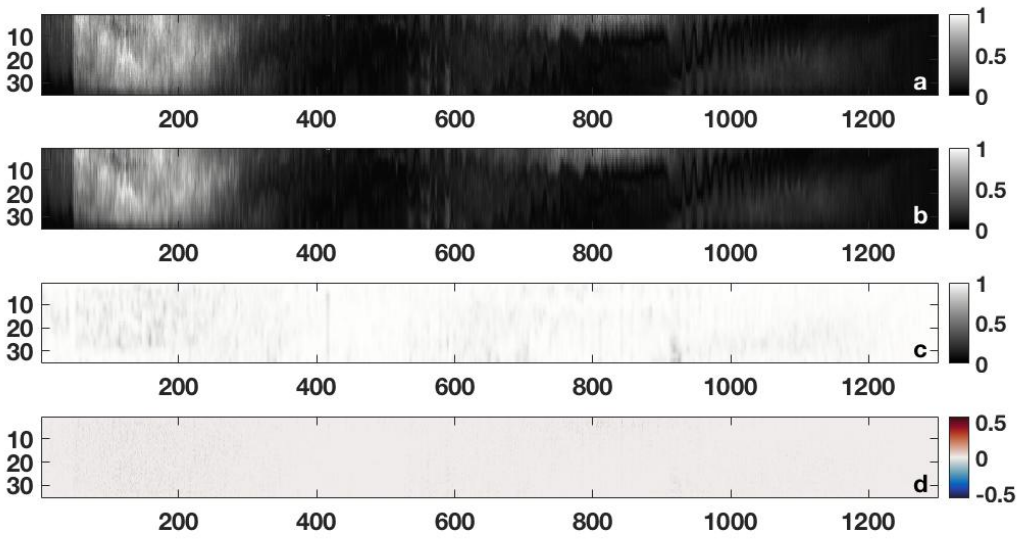

Figure A.5: The results for the optimal SVHT 13 EOF reconstruction. The z and x axes display the pixel numbers. From top to bottom the panels are: data (a), 13 EOF optimal SVHT reconstruction (b), SSIM map of the data against the reconstruction (c), and the pointwise error of the data against the reconstruction (d). The panels a-c are in grayscale, and panel d uses *cmocean* balance. See the text for details.

If D is chosen using optimal SVHT the result is Figure A.5. Panel a displays the `mat2gray` image of the full data set, equivalent to a $D = 35$ mode reconstruction. The `mat2gray` image of the optimal SVHT reconstruction is displayed in panel b. Note that both panel a and panel b's colorbars have the same bounds, namely the minimum and maximum values over all EOF reconstructions. These same colorbar bounds are employed in panels a and b of Figures A.6 and A.7 as well as all panels of Figure A.8, so that comparisons between figures can be easily made. Panels a and b of Figure A.5 look nearly identical, as the nearly white SSIM plot of panel c shows that most values are near 1. The SSIM maps colorbar bounds are zero to one in all panel cs in all three of Figures

A.5, A.6 and A.7 for cross Figure comparison. Finally, panel d displays the pointwise error, meaning the data of panel a minus the data of panel b yields panel d. The maximum error is of order 10^{-3} in this case. Note that the colorbar maximum of the pointwise error for all of Figures A.5, A.6, and A.7 are set to be the maximum absolute pointwise error of the 1 EOF reconstruction against the raw data. This scaling was chosen as we expect the 1 EOF reconstruction to have the largest pointwise error (see Figure A.6). For symmetry, the negative of this value was taken as the minimum of the colorbars for pointwise error in all three figures.

Figure A.5 clearly shows that an optimal SVHT reconstruction is nearly perceptually identical to the full data set using `mat2gray`. We found this to be the case over a variety of choices of subsets of the full data. In this way the optimal SVHT reconstruction acts as a kind of baseline for low perceptual error reconstructions in our example, as this choice of D tends to produce a reconstruction almost indistinguishable from the raw data. The pointwise error shows a small amount of error which the reader will struggle to detect in the reconstruction of panel b, although the SSIM plot details where to look. Note in this case $D/M = 13/35 \approx 0.37$, so that the EOF reconstruction is only 37% the size of the raw data, using the calculation from section A.2.2. This reconstruction captures approximately 99.9% of the variance. The MSSIM is 0.98 to two decimal places.

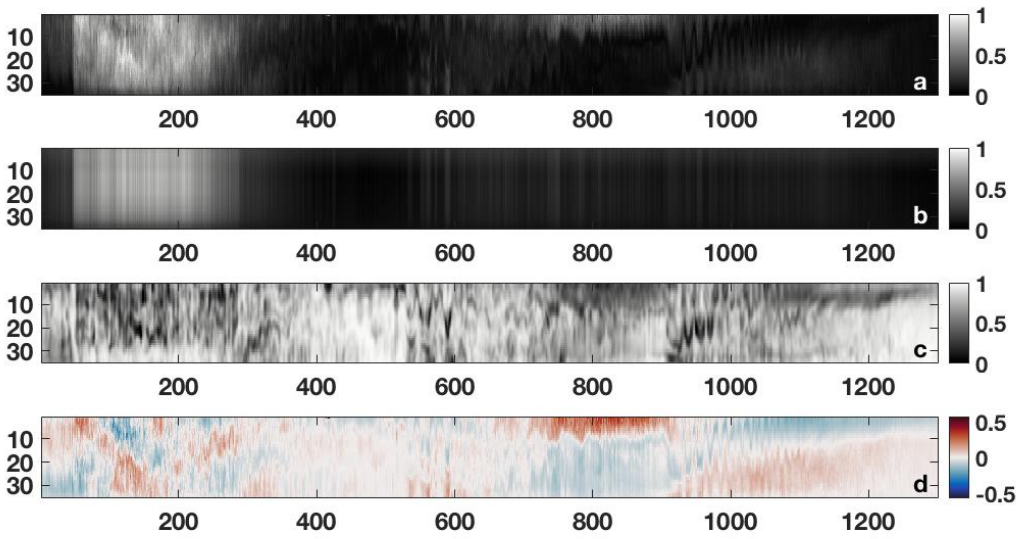

Figure A.6: The results for the 1 EOF reconstruction. The z and x axes display the pixel numbers. From top to bottom the panels are: data (a), 1 EOF reconstruction (b), SSIM map of the data against the reconstruction (c), and the pointwise error of the data against the reconstruction (d). The panels a-c are in grayscale, and panel d uses *cmocean* balance. See the text for details.

While optimal SVHT provides a nearly perceptually identical reconstruction, the 1 mode reconstruction has the most perceieved differences from the raw data among all choices of D. Figure A.6 has the same panels as Figure A.5, but using the 1 mode reconstruction with corresponding SSIM and pointwise error maps. The maximum error is of order 10^{-2}, worse than the optimal SVHT reconstruction. Unlike the optimal SVHT reconstruction, both the SSIM map (panel c) and pointwise error (panel d) have clear errors. The SSIM highlights the areas the eye notices as errors, while the pointwise error makes the mathematical structure of the error explicit. Clearly the 1 mode reconstruction misses a great deal of detail in the raw data, but still catches the first frontal crossing near the beginning of the record, as discussed in section A.2.1. If the application in question was only dependent on large changes in kinetic energy, perhaps this reconstruction would still be sufficient. This reconstruction takes up less than 3% of the space taken by the original data, and captures approximately 89.6% of the variance. The MSSIM is 0.72 to two decimal places.

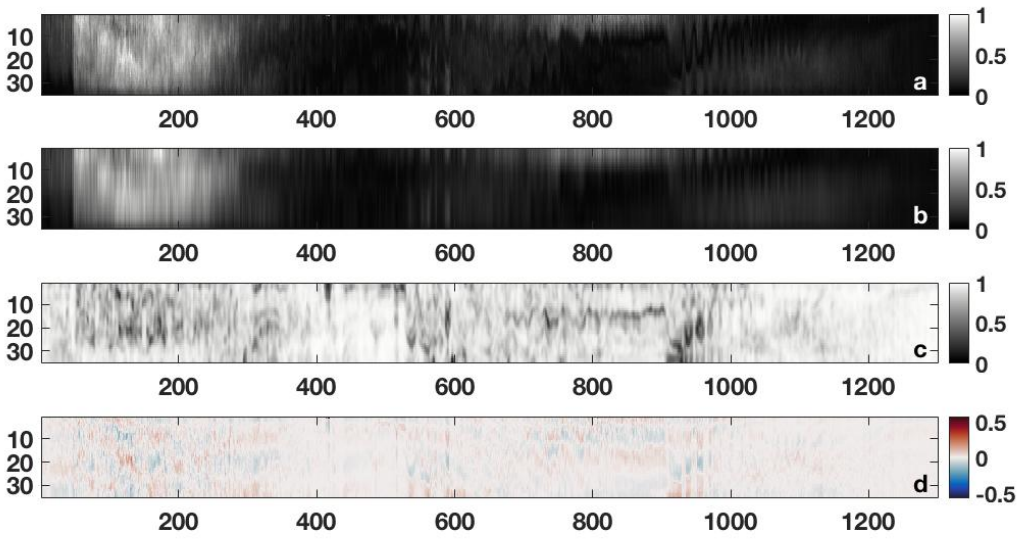

Figure A.7: The results for the 3 EOF reconstruction. The z and x axes display the pixel numbers. From top to bottom the panels are: data (a), 3 EOF reconstruction (b), SSIM map of the data against the reconstruction (c), and the pointwise error of the data against the reconstruction (d). The panels a-c are in grayscale, and panel d uses *cmocean* balance. See the text for details.

The reconstructions with D as 1 or as defined by optimal SVHT are two extreme cases. If some, but not all, of the features in the raw data are important, some intermediate value for D may be appropriate. If the presence of small scale structure, but not it's exact form, is important, the elbow test choice of $D = 3$ yields Figure A.7. The maximum error is again of order 10^{-2}, like the 1 EOF reconstruction case. It is clear that the reconstruction is missing details, but is an improvement over the 1 EOF reconstruction. The SSIM map (panel c) indicates that the data and reconstruction are closer than the reconstruction of $D = 1$, but less close than the choice of D by optimal SVHT, as expected. The SSIM map once more indicates the locations of perceptual error, and the pointwise error (panel d) outlines the mathematical error's structure. This reconstruction takes up less than 9% of the space taken by the original data, and captures approximately 98.3% of the variance. The MSSIM is 0.87 to two decimal places.

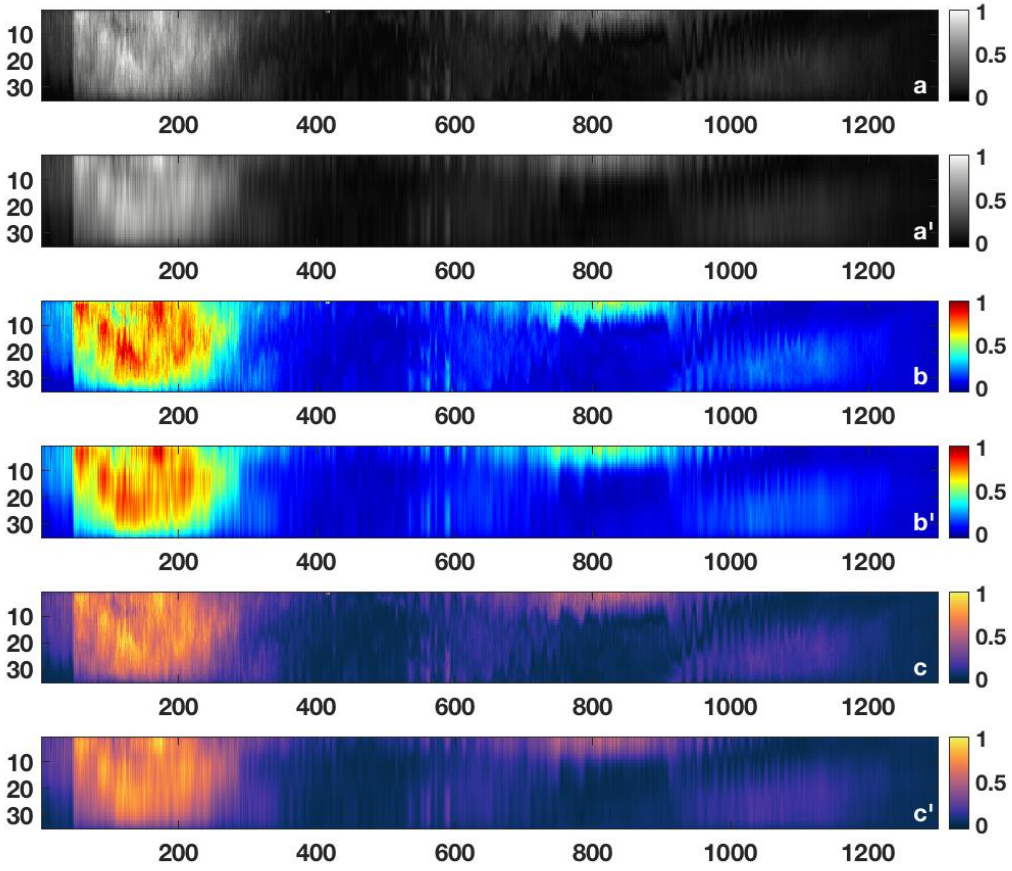

Figure A.8: The results for the 3 EOF reconstruction displayed using MATLAB's imagesc and 3 example colormaps. The z and x axes display the pixel numbers. The top two panels use *cmocean* gray: panel a is the raw data and panel a' is the 3 EOF reconstruction. The middle two panels use MATLAB's jet: panel b is the raw data and panel b' is the 3 EOF reconstruction. The bottom two panels use *cmocean* thermal: panel c is the raw data and panel c' is the 3 EOF reconstruction. See the text for details.

Let us assume that the 3 mode reconstruction is acceptable to the analyst in question. Figure A.8 gives three possible examples of their subsequent colormap choice. The six panels are paired with the raw data above the 3 EOF recontruction for each colormap.

From top to bottom, the colormaps are *cmocean* gray, MATLAB's jet, and *cmocean* thermal. The artificial gradients of jet, as illustrated in Figure 1e and 3 of [60], again manifest in the jet panels (b and b') of Figure A.8, making the perceptual difference between raw data and reconstruction more pronounced than in the perceptually uniform choices of the *cmocean* maps. Nevertheless all three examples show that the method presented here allows the tuning of the perceptual impact of data processing on the final perception of the visualization.

In summary choosing D by SVHT results in a near perfect visualization of the raw data, and choosing D by perceived feature degradation as quantified by SSIM allows tuning of the reconstruction to the desired application. Mode selection by optimal SVHT removes noise as defined by that framework, so it is perhaps expected that the visualizations of the optimal SVHT reconstruction would be nearly indistinguishable as measured by SSIM. EOF reconstructions using less modes than that prescribed by optimal SVHT have more perceived differences from the original data, with this effect getting more pronounced as $D \to 1$.

A.2.5 An SSIM False Positive

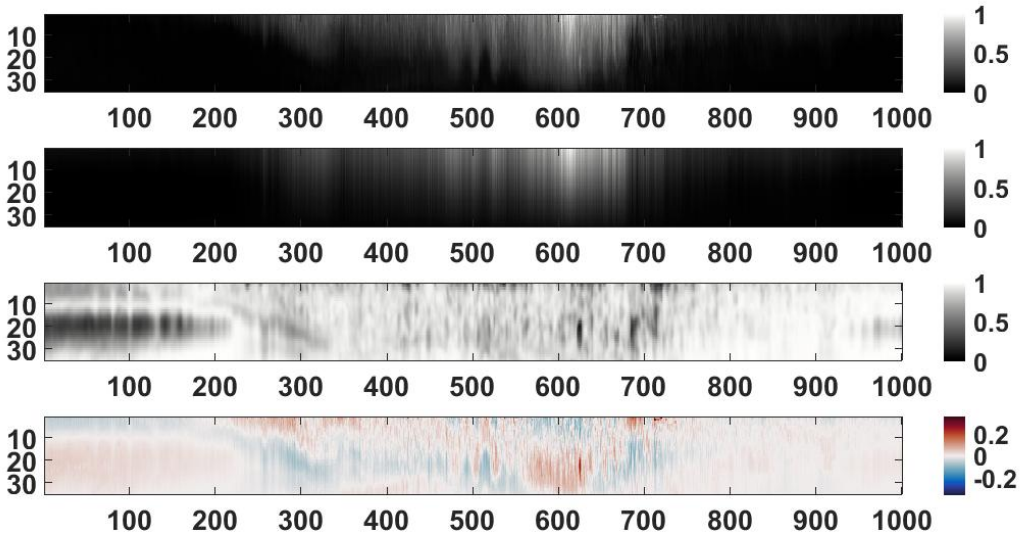

Figure A.9: The layout of this Figure and of Figure A.10 follow that of Figures A.5, A.6, and A.7 above. Clearly the extensive structure of the left side of the SSIM map is unwarranted.

Unfortunately, there is more to say on SSIM. On continued testing, we eventually found the example given in Figure A.9 which is an upwelling event corresponding to the gamma method $\mathcal{F}_1$ of Figure 2.2 panel d. In this case there is no perceptual difference in the leftmost 200 pixels or so of the 1 mode reconstruction, but the SSIM map shows a great deal of difference in this area. This is a case where the SSIM map fails to act as a perceptual norm, and instead gives a false positive.

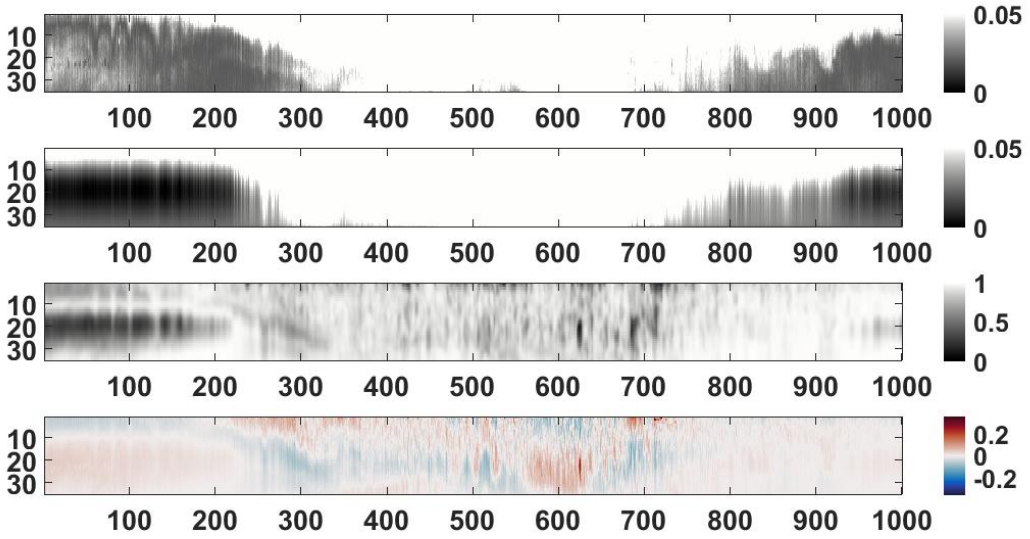

Figure A.10: The top two panels have been aggressively c-axised to make the source of the SSIM false positive evident. The bottom two panels are the same as those of Figure A.9

The source of the problem is evident in Figure A.10, where we have aggressively c-axised the top two panels. As outlined in section III C of [66], SSIM is a local measure. In this case it seems that the local small scale structure in the data causes the SSIM false positive. We had been using default values for MATLAB's `ssim` function. The formula for SSIM includes regularization constants: see equations 6, 9, and 10 of [66]. MATLAB allows you to set these constants, which are quadratic functions of a dynamic range variable by default. We explored the results of modifying dynamic range, and found that a value 500 times the default, corresponding to very large regularization constants, eliminated the false positive in the SSIM map. See Figure A.11. However these very large regularization constants also over-regularized the SSIM maps of higher mode reconstructions, washing out almost all detail. Removing the false positive had also made most of our SSIM maps useless. Perhaps there is some perfect choice of constants which would allow both the removal of the false positive and the retention of meaningful SSIM maps across a large range of D, but at this point we reconsider our approach. While we have found a choice of regularization constants which could avoid this particular false positive, it is not possible to say that there is a choice of constants which could avoid all false positives, or do so in a way which would avoid over-regularizing the SSIM maps of too many reconstructions. This means that anyone following the ideas presented here

would have to manually check the SSIM maps for all reconstructions to ensure there are no false positives. Even a single false positive requires different regularization constants to be chosen, and the entire algorithm to be re-run, so that all SSIM maps are with reference to the same set of constants. As we saw, this can introduce another problem, where choosing regularization constants which remove a false positive for one reconstruction can make the SSIM maps for other reconstructions featureless. All of this is cumbersome, and unlikely to be employed even if it can be made to work.

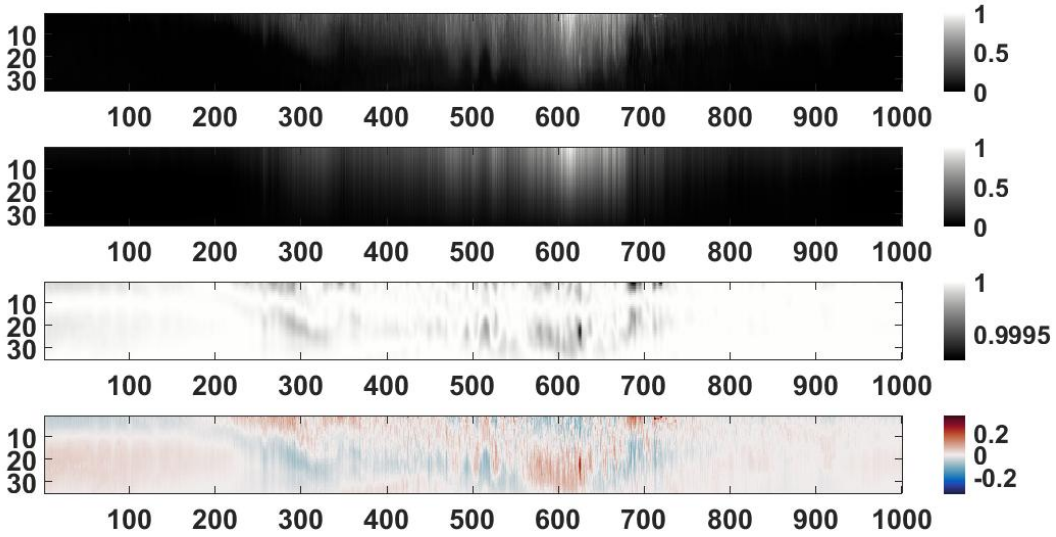

Figure A.11: The same situation as depicted in Figures A.9 and A.10, but with dynamic range set to 500 times its default value in the MATLAB implementation. This corresponds to much larger regularization constants than the default. Note the very narrow range of values for the SSIM map.

Our intention was to construct a method for measuring the perceptual effect of processing methods on visualizations of data sets. Until we encountered this false positive SSIM was a reasonable candidate to quantify perceptual differences: it was well-established in the literature, used in practice, and implemented in a variety of toolboxes. The work presented here showed promise. We were willing to look past the grayscale requirement, but the false positives are impossible to ignore, as they compromise the entire enterprise. At this point we decided that if the SSIM maps cannot be trusted, some other method of quantifying perceptual difference must be employed. This took us beyond standard

methods and implementations, and into a position where we would have to find or derive new methods, and write implementations. We were not convinced that such an enormous detour would be worth the significant effort. Instead we focused on feature identification. Nevertheless this appendix serves as an additional example of EOF in action. In particular it further illuminates the discussion on choosing D we began in section 4.3.5.