Machine Learning and Inference Methods for Surrogate Modeling and Inexpensive Characterization of Elastodynamic Systems

Thesis by
Alexander Charles Ogren

In partial fulfillment of the requirements for the
degree of
Doctor of Philosophy in Mechanical Engineering

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2024
Defended May 8th, 2024

ACKNOWLEDGEMENTS

First, I'd like to acknowledge my advisor, Chiara Daraio, for engineering a free-form research environment that allowed me a great deal of freedom in following my interests. My experience in grad school has been free of worry about where my funding will come from. This has been an incredibly meaningful source of stability, and enabled my mental bandwidth to be left available for much more interesting topics. Chiara has been a consistent guard-rail, allowing me to grow and learn at my own pace, and impressing upon me course-corrections for my studies and ways to improve the communication of my findings and research products.

I'd like to acknowledge my thesis committee - Domniki Asimaki, Cynthia Rudin, Katie Bouman, and Chiara Daraio - for all of the constructive interactions over the years, spanning research, teaching, and mentorship.

Katie Bouman has informally been a second advisor to me. Her mentorship has been critical to enable my exploration of topics beyond the traditional bounds of mechanical engineering, which has brought a refreshing dose of novelty and perspective-broadening to my PhD experience. From her, I've learned so much about how to navigate the details of research, efficiently solve problems, and direct the focus of my studies.

Berthy Feng has been a great collaborator and friend over the years. In research, she is easy to work with, and a sharp problem solver. At the brewery, she's a great conversationalist and a caring listener.

Jagannadh Boddapati has been an amazing colleague and friend. He and I have navigated our PhDs in parallel, sharing many grad school experiences, but more importantly trying new recipes, wading in rivers, and going on bike rides.

Working with Cynthia Rudin and Cate Brinson has been a truly enjoyable experience. From them, I've learned about how to pose useful and interesting research questions, how to demonstrate results and ideas in a clear and compelling way, and how to effectively guide students.

Working within a curious, analytical, clever, and creative community has been a major highlight of the last years. I'm grateful for friendships that have formed through collaboration with Zhi Chen, Mary Bastawrous, Han Zhang, Rayehe Karimi Mahabadi, and Jihoon Ahn, and discussions with Daniel Shahraki, Sharan Injeti, Gunho Kim, Brian Kim, Brian Ahn, Tracy Lu, and the rest of the Daraio Lab.

My experience was enriched by the Caltech staff. In particular, I would like to thank Carolina Oseguera and Jenni Campbell for making everything work out smoothly behind the scenes, and

Petros Arakelian for superhuman 3D printing turnaround times. The MCE shop staff have also been helpful to me, particularly Jon van Deusen and Trent Wilson. At the Caltech health center, Dr. John Tsai was very helpful to me, always going above and beyond, but especially of note when the pandemic began to complicate health center operations.

I'm grateful for the community at the University of Wisconsin - Madison, including Prof. Jacob Notbohm for giving me the opportunity to experience research as an undergrad, Profs. Wendy Crone and Sonny Nimityongskul and the Undergraduate Learning Center for giving me teaching opportunities as an undergrad, and all of my friends from UW.

To Aftermath Caltech Ultimate, thank you for the practices, tournaments, and all of the endorphins and friendships created along the way.

I am thankful for my friends, old and new, and my family, who have given me continual support, love, and joy. And to my partner, Faye, thank you for going on this journey with me.

ABSTRACT

This thesis has two main focuses: (1) surrogate modeling of elastodynamic systems, and (2) inference methods for the inexpensive characterization of elastodynamic systems. Elastodynamics is the study of how and why materials move and deform when they are subject to time-varying loads, covering a wide range of applications from architected materials, to telecommunications, seismology, sound isolation, non-destructive evaluation, and medical imaging. Here, we explore how to more efficiently model elastodynamics, and what we can infer about our environment from observing them.

The next generation of material engineering aims to leverage advanced multi-functional control over elastodynamic behaviors, but is currently limited by the large computational cost of purely physics-based modeling methods. Surrogate models aim to alleviate this cost by providing a data-driven approach to evaluate engineered material systems more efficiently. However, most current surrogate models lack certain useful traits, diminishing their potential for real-world use. This thesis begins by surveying the current state of surrogate modeling techniques, and establishes a set of state-of-the-art traits that greatly augment the utility of surrogate models, offering a perspective for the future direction of the field.

Next, a data-driven surrogate model based on Gaussian process regression for the computation of dispersion relations is developed, *GPR-dispersion*. The model exhibits several of the aforementioned traits, including representation invariance, data efficiency, incorporating direct use of physical theories, and the provision of both uncertainty estimates on its predictions and gradients for compatibility with gradient-based design optimization methods. GPR-dispersion is evaluated in comparison against both deep learning and traditional physics-based models.

The thesis then pivots to inference methods for the inexpensive characterization of material systems via partial observation of elastodynamic behaviors. Tissue stiffness is a tremendously important biomarker for a long list of health conditions, but often needs to be evaluated in a medical clinic with expensive equipment and highly trained workers. At-home health monitoring is a major next-generation goal of healthcare, but the trajectories of current consumer-grade sensor technology and biomarker inference methods have not yet fully intersected.

Inspired by a related work (*Visual Vibration Tomography*), *Visual Surface Wave Tomography* (VSWT) is proposed. VSWT observes partial information about the surface waves of layered elastodynamic systems (such as biological tissue) through monocular video to infer subsurface constitutive and geometrical information. Simulated experiments are presented to evaluate the accuracy, sensitivity, and limits of the method under ideal conditions. Real-world experimental results are presented using phantom materials that emulate biological tissue to demonstrate a practical proof of concept.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] Alexander C. Ogren et al. “Gaussian Process Regression as a Surrogate Model for the Computation of Dispersion Relations”. In: *Computer Methods in Applied Mechanics and Engineering* 420 (Feb. 15, 2024), p. 116661. ISSN: 0045-7825. DOI: `10.1016/j.cma.2023.116661`. URL: `https://doi.org/10.1016/j.cma.2023.116661` (visited on 04/14/2024).
A.C.O. conceptualized the project, wrote and revised of the manuscript (with greatly appreciated help from B.T.F, and suggestions from C.D. and K.L.B.), wrote the simulation code, and developed and tested the surrogate model. B.T.F. performed the comparison against Convolutional Neural Networks.

[2] Zhi Chen et al. “How to See Hidden Patterns in Metamaterials with Interpretable Machine Learning”. In: *Extreme Mechanics Letters* 57 (Nov. 1, 2022), p. 101895. ISSN: 2352-4316. DOI: `10.1016/j.eml.2022.101895`. URL: `https://doi.org/10.1016/j.eml.2022.101895` (visited on 04/26/2024).
A.C.O. contributed physical insights to the development of the method, developed specialized code for numerical modeling, designed and numerically validated the demultiplexer design demonstration, and participated in writing and revising the manuscript.

[3] Berthy T. Feng et al. “Visual Vibration Tomography: Estimating Interior Material Properties from Monocular Video”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 2022, pp. 16210–16219. DOI: `10.1109/CVPR52688.2022.01575`. URL: `https://doi.org/10.1109/CVPR52688.2022.01575` (visited on 04/19/2024).
A.C.O. contributed physical insights to the development of the method, developed numerical models in COMSOL for simulated experiments and damping characterization, contributed to the experimental validation, and participated in writing and revising the manuscript.

[4] Alexander C. Ogren et al. “Data-Driven Methods for Architected Materials: Data Acquisition, Property Prediction, and Intelligent Design”. In: *preparation* (2024).
A.C.O., B.Z., R.W., J.L. contributed equally to this work. A.C.O. led the analysis and writing of the "Forward Modeling" section, and wrote the version of the "Data for Machine Learning" section that appears in this thesis, which may be different from the final published version. B.Z. led the analysis and writing of the "Inverse Design and Optimization" section.

[5] Alexander C. Ogren et al. “Visual Surface Wave Tomography: Inferring Subsurface Material Properties from Monocular Video”. In: *preparation* (2024).
A.C.O. conceptualized the project, wrote and revised the manuscript, wrote the simulation code, and developed and tested the method. B.T.F. wrote and operated the motion extraction code. J.A. assisted with construction of experimental setup, and led the rheometry experiments.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 Research objectives

This thesis explores computational methods based on machine learning, computational imaging, and inference to advance the design and characterization of elastodynamic systems. There are two primary goals of the research in this thesis: (1) the efficient modeling of elastodynamic systems via data-driven surrogate models with useful traits like interpretability, incorporation of physics-based knowledge, and invariance to data representation; and (2) inferring the interior material properties of elastodynamic systems using incomplete observational data via computational imaging and inference methods. The first goal pertains primarily to the computational design of materials with engineered elastodynamic behaviors, aiming to open possibilities for the design of complex and innovative elastodynamic devices. The second goal fixes its sights on inferring the interior mechanical properties of material systems by partially observing the dynamics of the system's surface with monocular video, opening new possibilities for inexpensive material characterization methods, creation of virtual dynamic reprentations of environments or objects, and affordable at-home health monitoring.

1.2 Background and motivation

Elastodynamics is a branch of continuum mechanics that studies the behavior of elastic materials under dynamic conditions. When subject to time-dependent loading conditions, materials exhibit vibration and wave propagation. At present, the study of these phenomena is critical for applications in signal processing [1–3], seismology and geophysics [4–8], structural analysis [9], material characterization and imaging [10–12], and even medical treatment applications like breaking apart kidney stones [13]. For the future, researchers are currently studying the potential role that elastodynamics could play in applications of architected materials [14–17], expanding the scope of stimulation-based medical treatments to Parkinson's Disease [18–21] and targeted drug delivery [22], acoustic energy harvesting [23], and even quantum computing [24, 25]. The study of elastodynamic systems is relevant at a huge range of scales, from the small scales of cellular biology, ultra-compact mechanical signal processing components in smartphones, and the future of quantum computing, to the massive scales of structural analysis and earthquake engineering. Of particular relevance to this thesis is the elastodynamic modeling and engineering of architected materials, and the observation of the elastodynamic behaviors to image the interiors of objects and environments.

1.2.1 Advancing data-driven surrogate models for the design of architected materials

Architected materials (also known as *structured materials* or *metamaterials*) are materials with a mesoscale structure designed to yield unique and useful material properties. These mesoscale structures exist above the atomic and molecular scale, but below the scale of the macroscopic components that comprise the architected material. This mesoscale architecture, therefore, can exist at a wide range of scales. Since wave dynamics are so sensitive to the scales of features within their medium, this freedom of scale enables a tremendous amount of flexibility in designing architected materials to elastodynamically behave in prescribed ways for specific types of excitations. By thoughtfully engineering structured materials, we can exert a much greater level of control over fundamental wave behaviors and properties like refraction, reflection, diffraction, polarization, interference, phase, and dispersion, and can even achieve behaviors not found in standard materials like negative refraction [26]. In turn, controlling these fundamental behaviors can be leveraged to engineer more complex effects like lensing, mode and frequency filtering, anisotropic propagation, wave guiding, and signal multiplexing.

While the elastodynamics of architected materials have found many applications within academic research, real-world applications are few and far between. However, this is not for lack of ideas, and many possible real-world applications have been suggested, including the improvement of medical imaging and the improvement of mechanical signal processing in mobile telecommunications devices (including smartphones). At present, there are a few factors contributing to the lack of architected materials we see in the real-world. First, the manufacturing of architected materials, especially those designed with fine mesoscale structures, is not scalable. While fabrication technology is improving, the economics of manufacturing architected materials proves to be the limiting factor in some cases. Second, designing architected materials for specific physical behaviors is challenging. While interesting and compelling behaviors have been shown academically, many real-world applications require a high level of multi-functionality (e.g. requiring behavior X at frequency A, but behavior Y at frequency B, etc). Ultimately, architected materials exist in an insurmountably large design space, and navigating this space to determine the perfect architected material for a specific application remains a non-trivial task, especially if the feasibility of fabrication is taken into account.

Scientists and engineers have begun to employ state-of-the-art machine learning methods to aid with this complicated design problem. Machine learning tools are useful for learning complex relationships from data. In the context of architected materials, they are most often used to learn the relationship between the way an architected material is designed, and the effective properties that the design yields. This effort has three major fronts: (1) the curation of data from which to learn this relationship (2) the development of efficient-to-evaluate surrogate models that learn this relationship in the forward direction, often to be paired with design optimization methods, and (3) the use of generative machine learning that learn the relationship in the inverse direction, to produce designs that satisfy certain requested properties. Though each of these fronts will be discussed, this thesis is primarily focused on the second: the surrogate modeling for the efficient evaluation and design optimization of architected materials.

Traditional forward models as referred to in this thesis, will be defined as models based entirely on physical principles. They often enjoy favorable convergence behaviors, are highly interpretable, and are extremely flexible to operate on a wide range of scenarios, able to work with many different governing equations and a huge range of complex and disparate geometries. While extremely valuable as the backbone for many avenues of scientific inquiry and engineering solutions, they can be quite costly to solve.

A *surrogate model* approximates the input-output relationship of a traditional forward model with the goal of shedding an undesirable trait of the traditional forward model (often the large computational cost). Basic surrogate models have long been in development, but they often shed many of the extremely useful qualities of the physics-based models. They often depend on specific design paradigms, and are not designed to operate outside of these paradigms, massively limiting their flexibility to generalize to new scenarios. Further, many basic surrogate models, especially those based on deep learning, embody "black-box" input-output relationships, and therefore lack both interpretability and an incorporation of physics-based knowledge. The first major focus of this thesis analyzes these issues, and and demonstrates ways to incorporate the desirable qualities of physics-based models back in to surrogate models. In Chapter 2, I discuss the current state of data-driven surrogate modeling for physical applications in the context of architected materials, giving a more in depth analysis of these current challenges, offering perspectives for the future direction of the field, and highlighting a few existing surrogate modeling approaches that are already headed in this direction. In Chapter 4, I propose a surrogate model based on Gaussian process regression for the computation of dispersion relations that incorporates physics-based modeling, interpretability, and representation invariance.

1.2.2 Inference methods for the inexpensive characterization of elastodynamic systems

Elastodynamics are also extremely relevant to inferring characteristics of our environment and objects in our environment that would otherwise be difficult to know. This happens all over the place in nature, from the use of acoustics for echo-location in bats [27], dolphins [28], and whales [29] to the localization of prey and detection of predators in Spiders via elastic vibratory cues [30, 31]. Elephants have been shown to interpret information carried in low frequency geophysical waves both to communicate with each other and detect potential threats [32–35], and it is suggested that some primates use percussive techniques to locate cavities in wood in search of prey [36].

Waves and vibrations carry rich information about our environments. A biomimetic human adaptation of echo-location, reflection-based methods like SONAR enable the characterization and navigation of environments in situations where vision may be obscured. Ultrasonic methods are used to image the internal elastic properties of objects for medical imaging applications [10, 11], material characterization [37], and for the non-destructive evaluation of structures [38]. Seismic wave methods enable us to understand the material properties of subsurface geological features [5–8, 39].

In the context of medical imaging and the characterization of biological materials, it has been shown that tissue stiffness is a critical biomarker that can indicate the presence of a wide range of conditions, including cancer [40, 41], fibrosis both of the liver [42–44] and lungs [45–47], chronic obstructive pulmonary disease [48, 49], and cardiovascular conditions like hypertension [50–52] and artherosclerosis [53, 54]. Tissue stiffness can also be used to track the progression of degenerative diseases like osteoarthritis [55] and muscular distrophy [56, 57], providing guidance to treatments and interventions. Medical imaging techniques are well-developed to measure tissue stiffness, including ultrasonic [10], Shear Wave Elastography [11] and magnetic resonance [12] techniques. These methods can provide incredibly useful biomarker information, but often require thorough scans performed at medical centers using expensive equipment and highly trained personnel. Because of this, the economics of providing this kind of health monitoring to the entire population is currently economically intractable.

As the age of big data, computational inference, and affordable biosensors continues to develop, however, there are many who believe that wearable technology could play a significant role in the future of early detection and preventative care [58–61]. Wearable technology has already revolutionized the monitoring and treatment of diabetes with continuous glucose monitors (CGMs) [62] and insulin pumps [63]. Wearable cardioverter defibrillators can help detect life threatening ventricular arrhythmias, and even intervene with difibrillation [64] for people who are known to be at risk. Even beyond medical-grade wearables, sensors in exercise-based or daily-use devices such as smartwatches, smartrings, and smartphones can provide mountains of biometric data to help with health tracking. These devices can deliver near-constant measurements of heartrate, sleep, blood oxygenation, and temperature. Tracking these trends can deliver useful insights about whether a person should see a doctor, or provide quality-of-life improvements. Of particular note, the Oura Ring [65] uses constant biometric measurements, providing a variety of insights including menstrual tracking. When coupled with an app like Natural Cycles [66] (the first, and currently *only*, FDA approved birth control app), this data can even inform a woman about her monthly fertility cycle, which she can then use to advise her sexual activities [67, 68]. Although using menstrual tracking as birth control is not a novel idea (women have been tracking and using this information themselves even prior to the digitization of the idea), the automated, data-driven approach to this method greatly increases its effectiveness, and alleviates the burden of manual record-keeping. Overall, this technology offers a compelling alternative to hormonal or IUD approaches to birth control.

While some academic research has demonstrated compact wearable devices that can measure tissue stiffness [69] via on-skin ultrasonic methods, none are currently commercially available. A significant practical aspect of the problem is getting people to wear the sensors. Although small ultrasonic transducers could be incorporated into smartwatches and smartphones in the future, current version of these devices only contain low-frequency haptic actuators (tens to low thousands of Hertz), and real-estate for additional hardware in smartphones and wearables is often scarce. Fortunately, the low-frequency behavior of biological materials can tell us a lot about it characteristics. We believe that coarse information about tissue stiffness can be inferred using hardware that is *already available* in consumer-grade electronics. Our vision for this technology is to provide low-frequency excitations using haptic engines already existing in smartwatches, and visually observing the tissue response with smartphone camera systems.

In this thesis, we develop two methods to infer the material properties of elastodynamic systems through the partial observation of their surface dynamics with monocular video. The first method, *Visual Vibration Tomography* (VVT, chapter 5), assumes the object in question has a known geometry, and extracts partial information about the modes of vibration from the video, then uses a carefully crafted optimization formulation to infer the inhomogeneous distribution of the relative densities and elastic moduli throughout the full three-dimensional volume of the object. Simulated experiments are presented on a different geometries, exploring the limits and nuances of the method in a controlled environment. Real-world experiments on two-dimensional membrane systems with hidden acrylic defects and three-dimensional Jell-O cubes with hidden internal clay features demonstrate the real-world feasibility of the method. The second method, *Visual Surface Wave Tomography* (VSWT, chapter 6), assumes the object in question is a layered material system, and extracts partial information about the surface waves supported by the object. After processing this information into a partial observation of a dispersion relation, VSWT then uses optimization to infer the elastic moduli and layer thicknesses of the system. Simulated experiments are presented on several different layered systems, exploring the limits and nuances of the method in a controlled environment. Real-world experiments on a three-dimensional Jell-O and acrylic layered system are presented to demonstrate the real-world proof of concept.

1.3 Code repositories

A series of open-source code respositories was developed during the undertaking of this research, primarily for the computation of dispersion relations via the finite element method, and via the *GPR-dispersion* surrogate model. I wrote these repositories to be easily useable by others, extendable to new applications, and reasonably well optimized for performance. I encourage the use of these libraries by anyone and everyone who wants to use them, but ask that you credit me appropriately (ideally via Github's "cite this repository", and/or via citing this thesis or the appropriate related work).

Because I intend to continually expand the capabilites and features of these repositories over time, I won't delineate their functionalities in too much detail, nor provide documentation for them in this thesis. To check on their current functionalities and documentation, please visit the repositories at the following links.

2D-dispersion

Currently a Matlab library for finite element computation of the dispersion relations of two-dimensional pixelated unit cells (or representative volume elements).

`https://github.com/aco8ogren/2D-dispersion`

2D-dispersion-GPR

Currently a Matlab library for using the *GPR-dispersion* surrogate model for the computation of dispersion relations of two-dimensional pixelated unit cells (or representative volume elements). Dependent on *2D-dispersion*.

`https://github.com/aco8ogren/2D-dispersion-GPR`

3D-dispersion-matlab

Currently a Matlab library for finite element computation of the dispersion relations of three-dimensional pixelated unit cells (or representative volume elements).

`https://github.com/aco8ogren/3D-dispersion-matlab`

3D-dispersion-GPR

Currently a Matlab library for applying the *GPR-dispersion* surrogate model to the computation of dispersion relations of three-dimensional pixelated unit cells (or representative volume elements). Dependend on *3D-dispersion-matlab*.

`https://github.com/aco8ogren/3D-dispersion-GPR`

Though the following two repositories are more specifically tailored to the needs of my research projects, and therefore may not be as generally useful to a wide audience, code used for the development and operation of *Visual Surface Wave Tomography* can be found at the repositories below.

tissue-dispersion

Currently a Matlab library for computing the dispersion relations of unit cells (or representative volume elements) with periodicity in one direction, and with a fixed (homogeneous dirichlet) boundary condition along the bottom edge of the domain. For me, this was built to efficiently compute the dispersion relations of the layered systems studied in the development *Visual Surface Wave Tomography*, a tool critical to making the fitting problem computationally feasible.
`https://github.com/aco8ogren/tissue-dispersion`

tissue-characterization

Currently a Matlab library acting as the "main" repository of the *Visual Surface Wave Tomography* method. Currenty private, may become public.
`https://github.com/aco8ogren/tissue-characterization`

1.4 Outline of chapters

This thesis can be thought of in two major topics, though there is signficant overlap in the subject matter between the two topics. The first topic is *surrogate modeling of elastodynamic systems*, and is discussed in chapters 2, 3, and 4. The second topic is *inference problems for the mechanical characterization of elastodynamic systems*, and is discussed in chapters 5 and 6.

Note on terminology

In this thesis, the terms *metamaterials*, *architected materials*, and *structured materials* will be used interchangably.

Permissions

Permissions for the inclusion of published manuscripts are detailed below.

[1] Alexander C. Ogren et al. “Data-Driven Methods for Architected Materials: Data Acquisition, Property Prediction, and Intelligent Design”. In: *preparation* (2024).

The copyright is held by myself and the co-authors, as this work is not yet published.

[1] Zhi Chen et al. “How to See Hidden Patterns in Metamaterials with Interpretable Machine Learning”. In: *Extreme Mechanics Letters* 57 (Nov. 1, 2022), p. 101895. ISSN: 2352-4316. DOI: `10.1016/j.eml.2022.101895`. URL: `https://doi.org/10.1016/j.eml.2022.101895` (visited on 04/26/2024).

Permission from Extreme Mechanics Letters to re-use my own material in new works or for scholarly purposes is not required (with full acknowledgement of the published version of the article). Permission was obtained from first author, Zhi Chen, to include the manuscript in this thesis.

[1] Alexander C. Ogren et al. "Gaussian Process Regression as a Surrogate Model for the Computation of Dispersion Relations". In: *Computer Methods in Applied Mechanics and Engineering* 420 (Feb. 15, 2024), p. 116661. ISSN: 0045-7825. DOI: `10.1016/j.cma.2023.116661`. URL: `https://doi.org/10.1016/j.cma.2023.116661` (visited on 04/14/2024).

Permission from CMAME to re-use my own material in new works or for scholarly purposes is not required (with full acknowledgement of the published version of the article).

[1] Berthy T. Feng et al. "Visual Vibration Tomography: Estimating Interior Material Properties from Monocular Video". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 2022, pp. 16210–16219. DOI: `10.1109/CVPR52688.2022.01575`. URL: `https://doi.org/10.1109/CVPR52688.2022.01575` (visited on 04/19/2024).

Permission from CVPR to re-use my own material in new works or for scholarly purposes is not required (with full acknowledgement of the published version of the article). Permission was obtained from first author, Berthy Feng, to include the manuscript in this thesis.

[1] Alexander C. Ogren et al. "Visual Surface Wave Tomography: Inferring Subsurface Material Properties from Monocular Video". In: *preparation* (2024).

The copyright is held by myself and the co-authors, as this work is not yet published.

Chapter 2

DATA-DRIVEN METHODS FOR ARCHITECTED MATERIALS: DATA CURATION, PROPERTY PREDICTION, AND INTELLIGENT DESIGN

[1] Alexander C. Ogren et al. “Data-Driven Methods for Architected Materials: Data Acquisition, Property Prediction, and Intelligent Design”. In: *preparation* (2024).

Author contributions

A.C.O., B.Z., R.W., J.L. contributed equally to this work. A.C.O. led the analysis and writing of the "Forward Modeling" section, contributed to the writing and revisions of the introductory matter, and wrote the version of the "Data for Machine Learning" section that appears in this thesis, which may be different from the final published version. B.Z. led the analysis and writing of the "Inverse Design and Optimization" section.

Chapter preface

This section provides a review of the current state of research in data-driven methods for the design of architected materials. Of particular relevance to this thesis, it explores surrogate modeling paradigms for the efficient modeling of physical systems. In this context, it establishes the challenges faced by data-driven surrogate models, common shortcomings of the majority of current surrogate models, and distills a set of "desirable attributes" from this discussion to guide the future development of surrogate models, highlighting a few existing state-of-the-art surrogate models that currently exhibit these traits. Overall, this chapter depicts the current state of the field, and develops context for the motivations behind the engineering of specific attributes and characteristics into the data-driven surrogate models discussed in chapters 3 and 4, such as *generalizability*, *representation invariance*, *ease of set-up and use*, *transparent and principled modeling*, and *inverse-design utility*.

2.1 Abstract

Architected materials, or metamaterials, are engineered materials that leverage thoughtfully crafted internal structures to support exceptional and exotic functionalities. They have been studied in nearly all major physical domains, including electromagnetics, optics, acoustics, and mechanics, for exotic properties such as wave filtering, cloaking, negative refraction, or negative Poisson’s ratio. As researchers continue to demand increasingly sophisticated functionalities, the design space of architected materials continues to expand. This renders conventional design approaches such as

topology optimization and Edisonian methods ineffective. The growth of data-driven methods, such as machine learning (ML) and artificial intelligence (AI) offers solutions that learn from existing data, that accelerates both the evaluation of material properties as well as the selection of the materials' internal structures. In chemistry, for example, data driven approaches have been used to design new molecular structures and compounds, by reducing the need for expensive simulations and time-consuming laboratory experiments. In this review, we discuss data-driven AI/ML methods for architected materials in the aspects of three interconnected pillars that constitute the research framework: (i) Data curation, (ii) Forward modeling, and (iii) Inverse design. For each, current, mainstream approaches and examples for architected materials are provided. We also discuss potential future research directions, where we first emphasize the interpretability and explainability of ML, and call for more involvement of physics in the data-driven approach. Furthermore, we propose our vision for future metamaterials research that a software suite be developed to achieve all design components with minimal human intervention. Finally, we point out required steps for proposed research endeavors.

2.2 Introduction

2.2.1 What are architected materials?

Architected materials (or metamaterials) are materials with a mesoscale structure designed to yield unique and useful material properties. These mesoscale structures exist at a scale above those of atoms and molecules, which determine the properties of the constituent materials that make up the metamaterial, but below the scale of the components where metamaterials find practical use. Metamaterials are already employed commercially, for example, in bicycle helmets. At the macroscale, the structure of a bike helmet is designed to conform to a person's head, and provide ventilation through centimeter-scale openings. At the molecular level, these helmets primarily consist of polystyrene molecules, which provide chemical and thermal stability. At the mesoscale, the structure of bicycle helmets is dominated by micron-scale cavities to achieve high impact absorption capabilities, and a lightweight design. Metamaterials have been studied and used in a variety of settings. They first appeared in the context of electromagnetics where they were used to manipulate electromagnetic waves. For example, metamaterials were able to achieve negative permeability and permittivity [70, 71], and negative refractive indices [26]. As a result, metamaterial can cloak objects, making them invisible in some electromagnetic frequency bands [72]. Metamaterials later began to make an impact in mechanics and thermodynamics, with a focus on manipulating mechanical and thermal behaviors of materials, from stiffness and density, to wave propagation, energy absorption, reprogrammability, and heat transfer. For example, metamaterials can yield non-standard static properties such as negative Poisson ratio [73] or shear-normal coupling [74]; elastodynamic behaviors such as the redirection of surface waves [75] or nonlinear focusing of waves [76]. Some mechanical metamaterials can act as shape-reconfigurable materials that undergo dramatic changes in shape and stiffness by leveraging particle-packing physics [77], multi-stabilities [78], or complex buckling behaviors [79]. Additionally, metamaterials have also been employed in multiphysics problems. For example, thermomechanical metamaterials can be designed to satisfy structural requirements for stiffness while minimizing the conduction of heat [80]. Because of their ability to attain useful and unique properties, metamaterials have the potential for high impact in fields such as telecommunications, medical imaging, vibration isolation, sustainable structures, reconfigurable/deployable structures, energy absorbing materials, and many more.

2.2.2 Challenges

The advantages of metamaterials arise from the carefully designed mesoscale structures. With the advance of additive manufacturing methods, which enable the fabrication of materials with ever higher structural complexity, a large, parameterized design space ensues, due to the infinite accessible geometries and material choices. For example, for pixelated or voxelated metamaterials, the number of design possibilities grows exponentially with the number of pixels or voxels. Additionally, parametrically defined geometries such as Bézier-curve based designs exist in a truly infinite design space. Not only are there endless configurations of parametric operations that can be combined to form a geometry, but each operation typically relies on one or more continuous parameters that can assume an infinite number of values. These examples exceed the computational limit of brute force approaches and other conventional design algorithms. Most previous designs for metamaterials were largely dependent on the designers' experience and on extensive trial-and-error efforts, which led to the creation of periodic architectures of repeating units (i.e., unit cells) with motifs derived from a limited number of known geometries. However, these approaches are generally slow, inefficient and often fail to select the best material candidates.

2.2.3 Necessity of using Machine Learning

Machine learning (ML), a branch of artificial intelligence (AI), leverages large and diverse datasets to learn patterns and make rapid property predictions, reducing the need for costly and time-consuming experiments or simulations. It also solves inverse design problems using deep generative models. The advancement of deep learning, particularly transformer models used in large language models (LLMs), has extended beyond natural language processing to complex applications like protein folding and drug design. In metamaterial design, integrating domain knowledge about mechanical and physical mechanisms into ML models enhances their reliability, robustness, and explainability. Despite widespread use in metamaterial research, there is a push towards developing comprehensive software that integrates data curation, forward modeling, and inverse design into a unified platform with minimal human intervention, as depicted in Figure 1.

In this review, we survey the state-of-the-art in data-driven AI/ML methods for metamaterials, covering three main pillars: data curation, forward modeling, and inverse design. For each pillar, we discuss current mainstream approaches and provide examples. We emphasize the importance of interpretability and explainability in machine learning, advocating for greater integration of physics in data-driven approaches. Furthermore, we propose a vision for future metamaterials research, suggesting the development of a comprehensive software suite that minimizes human intervention in the design process. Figure 1 illustrates the relationship and interaction between these pillars, forming the basis of our vision for future research directions in metamaterials. We also highlight the grand challenges facing this field and outline necessary steps to achieve the proposed research goals.

2.3 Data for Machine Learning

Data is the backbone of machine learning techniques. Within the context of architected materials, the primary goal of ML is often to learn the relationship between the design and effective properties of architected materials, and secondarily can be used to compactly represent architected material designs. The choice of how data is represented and generated can have a significant impact on the quality of an ML model built from that data. In this section, we discuss the common paradigms for these choices in the context of architected materials, and survey their benefits and detriments.

2.3.1 Data representation

In this section, we discuss and evaluate common data representations for architected materials, and some novel techniques that have been used to unify seemingly disparate data representations.

2.3.1.1 Parametric representations

Parametric representations of metamaterial designs are encoded by a typically low-dimensional set of parameters. Because they comprehensively describe the design with a small number of parameters they are more compact than other types of design representations. In exchange for this compactness, however, parametric representations are often locked in to specific design paradigms. As a result, these types of representations are limited to represent a relatively narrow slice of design space, and therefore are not considered to be very extensible.

In some cases, parametric design representations directly encode human-interpretable characteristics of the design, such as geometric or constitutive parameters. For example, a structured material may be described by the radii and center locations of a series of circular holes that define the mesoscale structure. While this description is quite compact, it is also quite narrowly scoped, as it cannot describe mesoscale structures that are not defined by a series of circular holes.

Alternatively, parametric design representations might contain less direct descriptors of their mesoscale architecture. For example, one might describe parameters that govern the likelihood of pre-defined geometric features appearing in a randomly generated structure. Although these types of representations are also compact, and can technically statistically span a larger design space, they relinquish a tremendous amount of design control. Further, they still rely on pre-defined mesoscale features, which ultimately restricts the types of designs they can represent.

Non-linear dimensionality reduction (NLDR) techniques offer a unique and clever trick to combine compactness with extensibility. Examples of NLDR techniques include isomap, locally linear embeddings, laplacian eigenmaps, kernel PCA, diffusion maps, autoencoders, and more. These techniques are specifically engineered to reduce the dimensions of high-dimensional data into a more compact form while preserving the intrinsic information contained within the data as much as possible. They work by compressing high-dimensional data into a *latent space*. In the context of structured materials, these techniques have been demonstrated to be able to describe metamaterial designs in a much more compact way, while still being able to represent designs that come from different parametric design paradigms. This offers an interesting and powerful way to combine datasets that use completely disparate design descriptors, and build a latent space where the union of the data can be described by a unified set of latent parameters [81, 82].

2.3.1.2 Discretized representations

Discretized representations of structured material designs offer the opposite balance of compactness to extensibility, as compared to parametric representations. They rely on a spatial discretization of the structured material, and are typically encoded by relatively large arrays that not only define the discretization, but also define the relevant constitutive parameters in each region of the discretization. Because of this, discretized representations tend not to be very compact. In exchange, they are often very extensible, being able to represent (at the very least in close approximation) an extremely wide variety of material designs. Examples of discretized representations include mesh-based representations (which pixel- and voxel-based representations, and STL representations are a subset of), graph-based representations, and point-cloud representations. Despite being extremely heavy, they offer a very comprehensive and principled description of the design space, and are used in the most computationally-impressive realizations of metamaterial design optimization [83].

Discretized representations do not preclude the use of NLDR techniques. In fact, when using NLDR to unify the description of disparate parametric design representations (as discussed in 2.3.1.1), parametric representations must first be converted to a universal description as an intermediate step before being sent to the NLDR algorithm. This universal description is always a discretized representation.

2.3.2 Data sources

As we know, the most fundamental requirement for using machine learning methods is that we must have data that represents a pattern. The central focus of metamaterial design is to understand the relationship between design and function. For this reason, if we are to use machine learning to assist with this design problem, it is critical to be able to generate large amounts of data that embody this relationship. In the context of machine learning, data is often described within the paradigm of *inputs* and *outputs*, often denoted as x and y. For each x, the machine learning tool is expected to learn the correct expected output y. In the context of metamaterials, this often takes the form of inputs d representing a metamaterial design, and outputs p representing the effective properties, behaviors, or functionalities that result from that design. Only by thoughtfully constructing datasets that contain relevant design-functionality patterns, can we effectively train machine learning models to help us design metamaterials. In this section we will discuss the methods for data generation, and the challenges associated with these methods.

2.3.2.1 Real data

Real data (or *empirical data*) is data that is measured directly from the real world, rather than from simulation. In the context of structured materials, real data would be collected from real-world experimental methods that measure the properties of known metamaterial designs. Practically, this kind of data can be quite difficult to obtain, because experimental methods require plenty of expertise and care to execute. Further, to obtain large amounts of data that represent a sufficient breadth of the design-functionality relationship, experimental data generation methods require a tremendous number of metamaterial samples to be fabricated. While this type of data can be costly to produce, it can also be quite valuable, since it is the only direct way to sample real-world physics. Especially in situations where the purpose of the machine learning task is to learn a constitutive relationship or physical law that is hard or otherwise evasive to pin down mathematically, empirical methods may be the only reasonable choice for data generation. There is no numerical simulation quite like real life.

2.3.2.2 Synthetic data

Synthetic data is data that is produced by simulation. We call it synthetic because it often does not accurately represent real-world relationships, lacking the nuances and details that might be contained in genuine empirical data. However, synthetic data is often significantly less costly to generate, and can help evaluate whether a certain machine learning method is valuable to pursue in the long run. Further, in some cases, complementing real data with synthetic data can be a very cost effective way to balance the need for real data with the need for lots of data. A final benefit of synthetic data, in comparison to real data, is that it can often be made noise-free, which is often not the case for real data. Noise-free data can offer a "best case scenario" limit, representing a ceiling for how well a machine learning method could possibly perform when trained on ideal data.

2.4 Forward modeling

2.4.1 Why model architected materials?

The ability to physically model architected materials is critical to their design and development for real-world use. Ultimately, this means understanding the relationship between the way the architected material is designed (for example, the geometry and constitution of the material) and the effective material properties that the architected material exhibits.

Typically, to understand this relationship, we define a function that takes a metamaterial design, d, as an input, and gives its material properties, p, as an output. A model that operates in this direction is called a *forward model*. Let us denote this mapping as $p = f(d)$. Forward models are generally functions, in the mathematical sense, because for each metamaterial design, there is a single set of properties that describe it. For example, a metamaterial cannot have two different effective densities. In the vast majority of cases, a forward model involves solving a *partial differential equation* (PDE) which mathematically characterizes the physics of interest. These equations could involve elasticity, plasticity, heat transfer, electromagnetics, or even a combination of these, among others.

In contrast, *inverse models*, which map effective properties onto metamaterial designs, are generally not functions (i.e. the forward model f is not an invertible function). For example, there are many different ways for a metamaterial to achieve the same effective density. However, we can still obtain reasonable solutions to inverse problems with forward models by wrapping them in optimization algorithms, which rely on a repeated evaluation of the forward model to converge on a design solution that achieves specific design goals. Because of the necessity for repeated evaluation, forward models that are computationally costly have a particularly cumbersome time working in tandem with optimization algorithms. In this section, we illustrate how machine learning has shown considerable promise in improving the efficiency of these forward models.

2.4.2 Conventional modeling of architected materials

There are two main focuses of modeling metamaterials:

1. *homogenization*, which is the characterization of a representative volume element (RVE), and

2. characterization of macroscale structures composed of metamaterials, which generally uses

homogenization to more efficiently model macroscopic designs made from metamaterials.

An *RVE* is a subdomain of a metamaterial that sufficiently represents the material as a whole. For this representation to be accurate, it is important for the RVE to be large enough to be statistically representative of the entirety of the mesoscale architecture. Note that in this section, we will use RVE as a term that applies to both periodic and aperiodic metamaterials; for periodic metamaterials, the RVE is equivalent to the unit cell. Because metamaterials rely on their mesoscale structure to yield interesting properties, physical simulations of them require these mesoscales to be well-resolved. In the case of macrostructures composed of metamaterials, it is often intractable to explicitly model every degree of freedom, so scientists and engineers typically turn to homogenization approaches.

Homogenization is a tool used to understand how an RVE behaves when it is part of a larger material assembly. Generally, this involves performing analysis on one RVE to mathematically derive the parameters of a reduced-order model for the overall behavior of the metamaterial that the RVE constitutes. As an analogy, this can be compared to using mathematical analysis of an atom to understand the larger behavior of a crystal lattice that it is a part of. An example of homogenization is the computation of an effective material stiffness tensor of the RVE so that it can be modeled with fewer degrees of freedom in the future. During homogenization, scientists often employ assumptions to reduce the domain of analysis to a single RVE (for example, the assumption of periodicity). These clever mathematical shortcuts allow us to avoid modeling the entirety of a metamaterial lattice (saving greatly on computational cost), while still gaining insight about how any given unit cell in that lattice will behave. Homogenization can be straightforward and inexpensive or complicated and costly depending on the nature of the problem at hand. Importantly, to use homogenization it must be possible to define an RVE of appropriate scale, and a composition that accurately represents the macroscopic material domain. While this is always possible for periodic metamaterials (where an easy choice for the RVE is the unit cell), defining an appropriate RVE can be trickier for aperiodic metamaterials.

Once the RVE has been characterized, scientists and engineers often turn to a homogenized framework for the engineering of the macrostructure. In this framework, each RVE of the macroscale structure is treated as a homogeneous domain, and is represented by the reduced-order model found by the homogenization process. Doing this allows us to massively reduce the number of degrees-of-freedom needed to model macroscale domains composed of architected materials.

2.4.3 Challenges with traditional modeling of architected materials

Often, the reason for modeling a metamaterial is to solve an inverse design problem using optimization, repeatedly evaluating and changing a material design until the material exhibits the desired properties. However, even with impressively efficient theoretical and numerical methods, cuttingedge computational techniques (including parallelization and hardware acceleration), and powerful approaches developed in the field of design optimization, computational limitations still prove to be a constraint in the modeling and design of metamaterials. For example, one of the most impressive undertakings of design optimization for phononic bandgaps implements a specially-tailored physics-based homogenization solver on a highly parallelized, GPU-accelerated, distributed computing framework [83]. Despite the formidable setup, the authors needed to limit the resolution of their studies (in some cases even enforcing design symmetries), and the complexity of their objective function (maximizing the width of a single bandgap) to make the study possible. Because the computational cost of the forward model is often a significant limiting factor, the use of ML to accelerate the forward modeling process holds great potential to expand the scope of current and future metamaterial design efforts.

2.4.4 ML surrogate models

To accelerate the forward modeling of metamaterials, ML can be used to build surrogate models. A *surrogate model* approximates a traditional forward model (which is usually entirely rooted in physics) with the goal of shedding an undesirable quality of the traditional forward model, such as high computational cost. In the context of metamaterials, we are typically interested in constructing a surrogate model, $\tilde{f}$, which can approximate the mapping $p = f(d)$ while incurring only a fraction of the computational cost required by the traditional forward model f.

Many ML-based surrogate models are data-driven, combining data, priors, and a fitting routine to form a hypothesis about the function they are trying to emulate. Thoughtfully choosing each of these components is important: the data must sufficiently represent the input-output relationship of the forward model, the priors must reflect reasonable assumptions, and the fitting routine must be able to navigate the model to an accurate and generalizable hypothesis.

2.4.4.1 Desirable attributes for surrogate models

All surrogate models are designed with fundamental goals in mind that are central to the justification of the surrogate model. For example, a surrogate model must be accurate enough for the problem at hand, and must be able to make evaluations more quickly than the traditional forward model it is replacing. Further, a surrogate model should never require so much data to train that the cost of producing data outweighs the benefit of using the model. Beyond these basic requirements, some surrogate models go above-and-beyond, and are designed to exhibit additional desirable attributes. These attributes are not necessary for all use cases, but can greatly add to the value provided by the surrogate model, as listed below.

1. ***Generalizability*** is the model's capability to apply learned knowledge to unseen data. Al-

though all surrogate models are expected to make accurate predictions for data in their training set, and even for unseen *in-distribution* data, well-designed surrogate models can make accurate predictions even for *out-of-distribution* data, i.e. data that was not well represented in the training set.

2. ***Representation invariance*** is the ability for a surrogate model to operate in a consistent way with data of various formats, accepting inputs in different representations and making predictions at arbitrary resolutions, rather than being locked in to specific representations or resolutions. This is highly useful for learning from data of various representations, and enables the model to be flexibly used across a wider variety of scenarios.

3. ***Ease of set-up and use*** measures the cost of the overhead required to train and deploy the model. Models that are quick to train and require less data to learn effectively can be more easily repurposed to new design tasks.

4. ***Transparent and principled modeling*** involves the use of sensible, understandable logic and/or adherence to physical principles within the model. Models that are designed with interpretability or physical-laws in mind lend themselves well to scenarios where the reasoning behind predictions is important, and can offer a higher level of assurance to the user of the model.

5. ***Inverse-design utility*** is the model's effectiveness in contributing to the design process. For example, surrogate models that can efficiently supply the gradient of their output with respect to their input, or provide uncertainty metrics on their predictions can be much more conducive to efficient explorations of the design space when paired with optimization algorithms.

2.4.4.2 Basic surrogate models

In this section, we describe the most common trends in the construction of surrogate models that we see in the literature. They are not so fancy, but they get the job done in simple situations, and are often very *easy to set-up and use*. Unfortunately, they often lack *generalizability, representation invariance, and transparent and principled modeling*.

Selecting predictors

Predictors are the features of the metamaterial design that the surrogate model uses as input to infer the material properties of the design. Selecting appropriate predictors is a key decision when constructing a surrogate model, and has a strong influence on both the performance of the surrogate model and its ability to generalize well in unfamiliar situations. The simplest choice of predictors is a set of variables that parametrically define the geometry. This choice generally allows for a much simpler model, but it also severely narrows the scope of the model, limiting it to interact only with designs that can be described by that same set of variables. A more general choice for predictors, pixel or voxel material values have the ability to represent a much wider array of designs, but often require more complex models, and models based on these predictors often do not generalize well to designs represented by alternative discretization schemes.

Parameter-to-property mapping

In the most basic form, ML can be used to directly learn the mapping between a set of design parameters and the effective material properties of interest. For example, the predictors d might be a radius, height, and elastic modulus that parametrically describe the design of a unit cell. Because the set of predictors is relatively small, this typically results in simple models that need only a small amount of data, and can be trained relatively quickly. These models are relatively easy to set up, and are excellent for learning one-off mappings for a specific design situation, but are not able to generalize to scenarios that depart from the specific predetermined parametric design paradigm used by the training data.

Discretization-to-property mapping

Using mesh-based design representations, for example pixel- or voxel-based representations, as predictors offer a much larger degree of generalizability. As previously discussed, pixels and voxels can represent metamaterial designs much more flexibly than parametric representations. When used as a predictor, d, the pixel/voxel values are typically passed to the surrogate model as a multidimensional array in an orientation that spatially represents the design. Because convolutional neural networks (CNNs) excel at capturing trends based on spatially-structured data, they have been extensively used as surrogate models for these types of predictors. These models are typically far more complex than parameter-to-property models, requiring larger training sets and longer training routines, but can operate over a wider range of scenarios.

For example, Finol et al. [84] demonstrated that a CNN can learn the function that maps two-dimensional pixelated unit cell designs to their dispersion relations, which describe the propagation of elastic waves in the metamaterial. The neural network inputs are formulated as arrays that represent the material properties at each pixel of the unit cell, concatenated with the wavevector, and the outputs are points on the dispersion relation. With sufficient training data, the model learns the relationship between the pixelated design of the unit cell, and the corresponding dispersion relation.

Even within this framework, there are limits to the generalizability of the model. For example, the relationship that is learned is specific to the discretization scheme established by the training set. If the model is applied to scenarios that deviate from this scheme (for example, if the arrangement, size, or shape of the pixels changes), the learned spatial interpretation of pixel values and their relationship to material properties is no longer relevant.

2.4.4.3 Advanced surrogate models

Advanced surrogate models go above-and-beyond to achieve two or more of the desirable properties of surrogate models. While they may require more critical thought and effort to develop, they offer a range of truly advantageous qualities.

Neural operators and operator learning
Typical neural network architectures are designed to learn mappings between finite dimensional spaces. While they can be effectively used to build surrogate models for fixed representations of metamaterials, they are not useful for building representation-invariant models that can consistently operate on data that uses a different discretization or representation scheme.

In contrast, a new paradigm of representation-invariant machine learning has been recently introduced, termed *operator learning*. One specific class of model within operator learning is the *neural operator* (NO). NOs extend deep learning from mappings between finite-dimensional vector spaces to mappings between infinite-dimensional vector spaces (function spaces). To provide this functionality, the authors replace the traditional finite-dimensional linear transformations between neural network layers (taking the form of matrix multiplications) with infinite-dimensional linear operators (taking the form of parameterized integral operators or multiplication in the spectral domain). In practice, even these infinite-dimensional, continuous operations must be implemented discretely in a computational setting, but when sufficiently discretized the neural operator exhibits resolution-independent error, and converges to a continuum operator in the limit of discretization refinement. This principled approach abstracts away the specifics of discretization schemes while maintaining important neural network properties such as universal approximation, producing a mathematically rigorous framework for operator learning. The application they consider is to solve PDEs by mapping the coefficient functions defining the PDE to the solution of the PDE. Because of the widespread use of PDEs in the modeling of a wide variety of physical phenomena, this is fundamentally relevant to the forward modeling of metamaterials.

They introduce four subclasses of neural operators, demonstrating that the fundamental mechanism of neural operators can be applied in a variety of modern neural network architectures: graph neural operators, multi-pole graph neural operators, low-rank neural operators, and Fourier neural operators, each of which offer model parameterizations designed to be efficient for modeling in different types of scenarios. For example, Fourier neural operators are parameterized to be efficient in scenarios where the problem can be represented and solved more efficiently in the spectral domain. Finally, they demonstrate the application of neural operators to solving well-known PDEs such as Burgers, Darcy subsurface flow, and Navier-Stokes. In the field of metamaterials, other researchers have extended the use of neural operators to build surrogate models for composite material systems [85], and even paired the neural operator model with optimization algorithms to inversely design photonic metamaterials [86].

Physics-based surrogate models

While many surrogate models learn mappings purely from data, some surrogate models have been designed to work together with traditional physics-based models, combining the benefits of learning from data with the grounding of principled, physics-based computations. Because these types of models use physics-based computations, they typically do not offer as steep of a speed-up factor as physics-agnostic models, but tend to exhibit a higher level of interpretability, and can often learn more effectively from smaller amounts of data than their black-box counterparts.

A physics-based surrogate model built on a Gaussian Process Regression (GPR) framework was recently introduced to accelerate the computation of dispersion relations [87], which describe wave propagation in materials. The authors combine a novel data-driven covariance function (the core element of most Gaussian Process-based models) with sparse measurements of the true underlying solution (supplied by a small number of physics-based computations) to infer the remainder of the solution. Because the model leverages physical computations, the time required for evaluation is larger than that required by most physics-agnostic models. In exchange, the model enjoys an elevated level of interpretability and physical basis, and a lower requirement for training data. The authors demonstrate the use of this model for the accelerated computation of dispersion relations for 2D unit cells.

The GPR-dispersion model never directly observes design variables d, and is only informed by them through the sparse physics-based simulations. Not only does this construction allow for the physics-based model to be modularly swapped for faster or more accurate physics-based methods that may be introduced in the future, but it also allows for representation invariance, since any metamaterial design can be modeled by some choice of a physics-based method. Further, because the model relies on a Gaussian process, it estimates predictive distributions rather than being limited to point estimations, and can therefore offer uncertainty metrics, indicating where and to what degree it is confident (or unconfident) in its predictions. In addition to representation invariance, physical-basis, and the provision of uncertainty metrics, the authors establish a mathematical framework for the GPR-dispersion model to provide gradients for compatibility with gradient-based optimization methods, offering *inverse-design utility*.

2.4.4.4 Challenges with convergence

Traditional physics-based models (such as FEA or other numerical methods) are generally expected to converge to an exact solution with mesh refinement. In contrast, most ML surrogate models exhibit a fixed (or at least limited), imperfect accuracy based on the size, resolution, and quality of the training data, the priors used to construct the model (such as network architecture, or other assumptions about the parameterization of the mapping), and the fitting methods used to train the model. These factors often cannot be changed after the model has been trained without incurring costs such as the computation of fresh, higher-quality training data and model fitting, both of which can be expensive processes. Because of these complications, ML surrogate models often do not offer promises of convergence to an exact solution in the same way that traditional physics-based models do. Some models such as the GPR-dispersion model and the neural operator model offer a tunable trade-off between the accuracy of the model and the cost of model evaluation, but even these models do not exhibit convergence to an exact analytical solution when implemented with practical computing limitations on the training data and model parameterizations. Fortunately, most design problems do not require exact solutions, and can tolerate some amount of approximation. For this reason, surrogate models can often still be reasonably used for metamaterial design despite their inability to exactly converge.

2.5 Inverse design and optimization

One major motivation for building surrogate forward models is to solve inverse design problems more efficiently. In the spirit of this, surrogate models are often designed to work in tandem with traditional optimization methods, providing important design information such as design gradients or prediction uncertainties (examples of *inverse-design utility*). Commonly, topology optimization methods rely on gradient-based approaches, where design variables are updated according to a gradient computation based on minimizing or maximizing a differentiable objective function. In some cases, especially when gradients may not be available, black box optimization methods including population-based methods like the genetic algorithm and statistically-based methods like Bayesian optimization have also been used. Statistics-based methods can often benefit from the provision of prediction uncertainties by surrogate models.

Alternatively, machine learning can be used for design in a more direct way. As previously discussed, while the forward mapping from design to property ($p = f(d)$) is often a function, f is often not invertible, meaning that there does not exist a $g = f^{-1}$ such that $d = g(p)$. This is for a few reasons. First, there is no guarantee that the requested material properties p are achievable by a design d. They simply might not be physically possible. Second, assuming they are possible, there may be many designs d that would yield properties p. Traditional optimization methods can handle this relatively easily by computing multiple solutions in parallel with different randomized initializations. However, traditional, function-emulating machine learning models cannot. Because of this, it is usually a better idea to use generative machine learning method, which learns statistic relationships rather than emulating a function. That is, a generative machine learning model, rather than (unsuccessfully) trying to learn $d = g(p)$, will (more successfully) try to learn the probability of a design given some requested properties, $\mathbb{P}(d|p)$. Mathematically, this allows for both the multiplicity of solutions, and the possibility that the requested properties p, if not physically realizable, do not result in *any* likely designs d. In this section, we will cover a few of the generative models most commonly used for inverse design of architected materials.

2.5.1 Deep generative models

Generally speaking, deep generative models (DGMs) compress the high-dimensional design input into a lower-dimensional latent space to represent and to eventually generate new material designs. Often using deep neural networks, DGMs model a joint probability distribution, aiming to learn a correlation between the training data and their labels. Once the generative networks are trained, the latent variable is fed into the network to which new data is generated as output. Among all DGMs, we highlight generative adversarial networks (GANs) and variational autoencoders (VAEs) along with their usage in metamaterial inverse design.

2.5.1.1 Generative Adversarial Networks

Consisting of two consisting networks, a generator and a discriminator, a GAN is trained when these two networks reach a Nash equilibrium: the generator learns how to generate new data that mimics training data and fools the discriminator, while the discriminator learns how to better differentiate the generated data from the training data, as illustrated in the first row of Fig. 4. Various GAN techniques have been used to inversely design metamaterials. An et al. presented a design approach for free-form all-dielectric metasurface devices combining a conditional generative adversarial network (CGAN) and a Wasserstein Generative Adversarial Network (WGAN) [88]. CGAN, a variant of GAN, can generate designs conditioned on class labels. As another variant of GANs, WGAN introduces the earth-mover distance to quantify dissimilarity, which measures the minimum amount of work to change one distribution into the other. The CGAN trains a generator that maps design conditions and a Gaussian noise vector to generate a target (fake) design, and a discriminator that calculates the earth-mover distance between the real and generated samples. The generator and discriminator are trained to minimize and maximize the earth-mover distance, respectively. Amplitude and phase profiles of EM waves were used as the design conditions, and the GAN was trained with EM responses from numerical simulations. Using this method, the authors demonstrated several multifunctional metasurfaces including a bifocal lens, a polarization-multiplexed beam deflector, and two multifunctional metalenses.

Shen and Buehler proposed a StyleGAN framework to design bio-inspired hierarchical metamaterials [89]. StyleGAN is a generative image modeling network that can control scale-specific features for designing hierarchical structures. This variant GAN model uniquely operates with the typical latent space of GAN and an intermediate latent space calculated from the typical latent space via a learning mapping network. Intermediate latent-space vectors are fed into the generator as input. GA was incorporated with a convolutional neural network (CNN) surrogate model that predicts effective material properties of the generated structure such as effective elastic modulus. The authors also showed that retraining the StyleGAN on the original dataset plus an additional dataset can broaden the design space. They demonstrated several material designs inspired by leaf microstructures via training their StyleGAN on a dataset of leaf micrographs. Jiang et al. developed a deep learning approach to bridge structural topology and the properties of elastic metamaterials, where the CNN and CGAN were employed for the forward and inverse problems [90]. Specifically, the authors focused on the prediction of dispersion relation for a given structural configuration and the proactive design of near-optimal structures based on the target dispersion relations. They showed that the developed framework is more likely to generate a feasible structure if the modes of the bands and their interrelationships are not substantially altered. They demonstrated material designs for bandgap engineering by re-customizing the existing dispersion relations based on band compression, stretching, and translation, as targets for the inverse design. The main principle of GANs, competition of the generator and discriminator, may lead to slow training. In addition, GANs may be unstable when the generator is only able to produce a small number of outputs.

2.5.1.2 Variational AutoEncoders

A VAE consists of an encoder that compresses input data (such as the metamaterial structure) into a lower-dimensional latent space, and a decoder that reconstructs a sampled variable from a distribution in the latent space to generate new data, as illustrated in the bottom row of Fig. 4. As metamaterial design examples, Wang et al. used a VAE to organize complex microstructures into a continuous and highly structured latent space [81]. They combined the VAE with a regressor for mechanical properties of interest, both trained on a metamaterial database. The generated metamaterials were integrated into the design of aperiodic and functionally graded multiscale metamaterials. Additionally, graph-based combinatorial optimization was employed to select the best microstructure and ensure compatibility between adjacent unit cells.

Wang et al. presented a deep learning framework for unsupervised generative design and optimization of cellular metamaterials for bandgap width and stiffness [91]. The authors used the Gaussian mixture beta VAE (GM-VAE) to extract latent features as design variables, which enables unsupervised clustering through DGMs. Gaussian Process (GP) regression models were also trained to predict the relationship between latent features and properties for property-driven design optimization. They adopted GA, with the GP model, to find the optimal latent variables. The generated designs were validated by simulations and replaced the low-performance designs in the training dataset, and the GM-VAE model was subsequently re-trained for the next iteration. They showed that their proposed designs outperformed ones obtained by topology optimization, in both stiffness and bandgap width.

Kudyshev et al. developed an optimization framework based on an extended adversarial autoencoder [92]. The authors used the conditional adversarial autoencoder (c-AAE) network where the encoder and decoder networks learn to compress and decompress input data while a discriminator network is additionally incorporated and trained to distinguish between samples from the compressed design space distribution and a predefined model. The design approach was applied to optimize thermal emitters for thermophotovoltaics utilizing GaSb photovoltaic cells. To rapidly estimate the performance of generated designs from the trained generator network, the c-AAE network is coupled to a conditional Visual Geometry Groupnet (c-VGGnet) that conducts the performance estimation based on the input binary image of antennas. Their optimized design exhibited 96% in-band emissivity and substantially suppressed the out-of-band emission, with lower computational cost compared to the previously demonstrated AAE-based local optimization frameworks. A major known drawback of VAEs is their tendency to produce blurry generative outputs even when they are trained using datasets with sharp boundaries, which arises from the formulation of loss functions. This may be unrealistic in material design problems.

2.6 Challenges and future directions

2.6.1 Interpretability and explainability of ML solutions

Often seen as a highly overparameterized black box, deep neural networks are notoriously difficult to interpret, making it challenging to learn design principles from design samples screened by a forward surrogate modeling and inverse designs generated by a DGM. Alternatively, while researchers have also resorted to intrinsically interpretable decision tree models in inverse design tasks [93], they are usually less powerful than deep learning models. However, despite the power of deep learning models, they have a generally acknowledged trade-off relation between accuracy and interpretability. For this, researchers have aimed at improving the interpretability and explainability of deep models for metamaterials by pursuing better mechanistic understanding and discovering interpretable key patterns within the metamaterial unit-cells related to properties of interest. Chen et al. proposed two concepts in the design of pixelized metamaterials: (1) shape frequency features, the occurrence frequency of certain shapes in the unit-cell; (2) unit-cell templates, arrangements of constituent materials in specific regions of the unit-cell [94]. Lee et al. utilized gradient-weighted activation maps to interpret the predicted sound absorption characteristics of metamaterials [95]. Such approaches fall in explainable AI (XAI) territory, which also include methods like SHAP (SHapley Additive exPlanations) [96], DeepLIFT (Deep Learning Important FeaTures) [97], and

LIME (Local Interpretable Model-Agnostic Explanations) [98]. XAI may help scientists and engineers unwrap the deep learning black-box and ultimately gain insights into what makes new metamaterial designs so effective.

2.6.2 Generalizability

Currently, much of the available data pertaining to the design of metamaterials is context-specific. For example, many datasets contain non-generalizable geometry definitions. Although ML models have been successful at learning relationships in these data independently, they often have trouble operating on data that is defined by different contexts. To enable truly large-scale, next-generation ML models (potentially with billions of parameters), it will be critical for scientists to converge on efficient, yet generalizable languages to describe metamaterial designs and their corresponding properties.

For this context-independent data to be useful, it is important to construct model architectures that leverage the data in principled and generalizable ways. Although the majority of existing models are simple-form models, focusing on learning a context-dependent mapping, some ML approaches have already started to work toward solutions to these problems, for example exhibiting representation invariance, incorporating physics-based computations, and describing metamaterial designs from different design paradigms in a unified latent-space design language. The continued development of data curation and data-driven methods within this paradigm is a cornerstone for the next steps in data-driven physical modeling and design.

Chapter 3

HOW TO SEE HIDDEN PATTERNS IN METAMATERIALS WITH INTERPRETABLE MACHINE LEARNING

[1] Zhi Chen et al. “How to See Hidden Patterns in Metamaterials with Interpretable Machine Learning”. In: *Extreme Mechanics Letters* 57 (Nov. 1, 2022), p. 101895. ISSN: 2352-4316. DOI: `10.1016/j.eml.2022.101895`. URL: `https://doi.org/10.1016/j.eml.2022.101895` (visited on 04/26/2024).

Author contributions

A.C.O. contributed physical insights to the development of the method, developed specialized code for numerical modeling, designed and numerically validated the demultiplexer design demonstration, and participated in writing and revising the manuscript.

Chapter preface

This chapter develops a surrogate model that exhibits *transparent and principled modeling* and *inverse-design utility*, two desirable attributes for surrogate models established in chatper 2, through the incorporation of geometric interpretability, and the ability to inexpensively directly sample designs from the surrogate model.

Here, we explore the development of two novel methods *shape-frequency features* and *unit cell templates*, which provide an interpretable framework for the surrogate modeling and elastodynamic design of periodic architected materials. The surrogate models are focused on predicting whether a band gap will exist within pre-specified frequency ranges, based on a binary 15-dimensional pixelated design space of stiff and soft pixels, conforming to the *p4mm* plane symmetry group. This approach allows an engineer to design band gaps into periodic material systems without repeatedly performing the expensive process of dispersion computations.

Each method is presented in detail, and evaluated against a variety of other ML methods. Finally, the unit cell template method is used to design a mechanical demultiplexer, which takes a multi-frequency input signal, and leverages band gap physics to split the signal into three frequency ranges. This design is validated numerically in a finite size setting with frequency domain analysis.

3.1 Abstract

Machine learning models can assist with metamaterials design by approximating computationally expensive simulators or solving inverse design problems. However, past work has usually relied on black box deep neural networks, whose reasoning processes are opaque and require enormous datasets that are expensive to obtain. In this work, we develop two novel machine learning approaches to metamaterials discovery that have neither of these disadvantages. These approaches, called *shape-frequency features* and *unit-cell templates*, can discover 2D metamaterials with user-specified frequency band gaps. Our approaches provide logical rule-based conditions on metamaterial unit-cells that allow for interpretable reasoning processes, and generalize well across design spaces of different resolutions. The templates also provide design flexibility where users can almost freely design the fine resolution features of a unit-cell without affecting the user's desired band gap.

3.2 Introduction

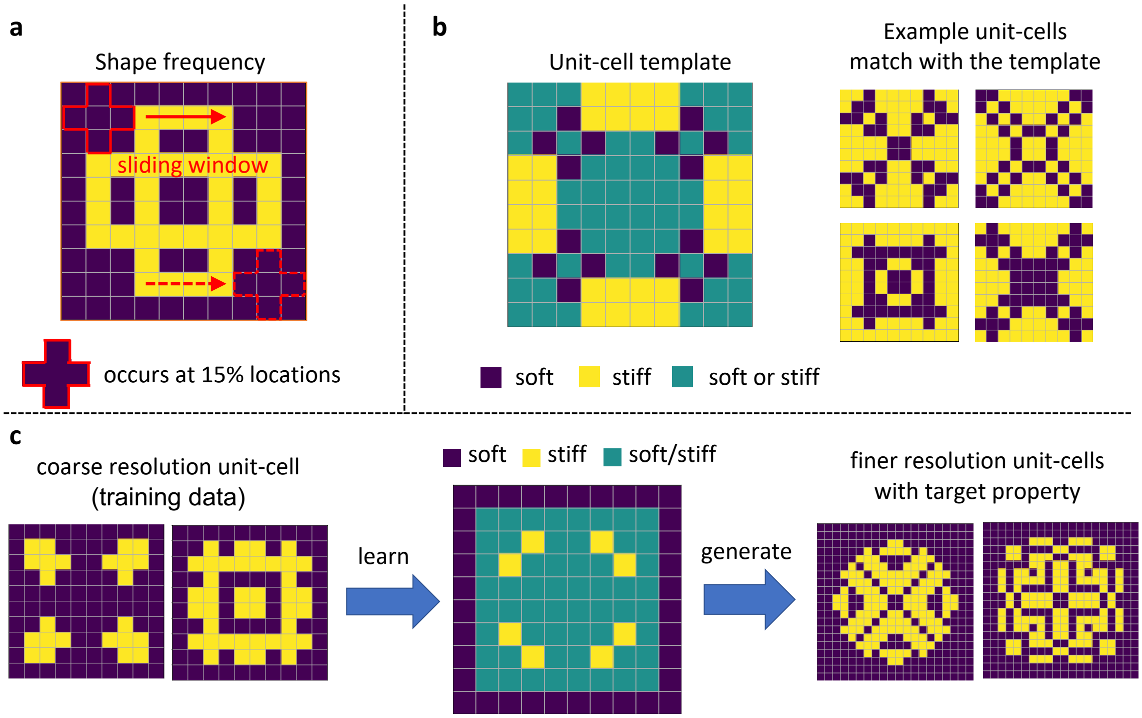


Figure 3.1: Examples of interpretable key patterns discovered by the proposed method. **a**. A shape frequency feature (this one is shaped like a "+"). How frequently this shape appears in the unit-cell is a useful predictor of a band gap. **b**. A unit-cell template, which considers specific global patterns in the unit-cell. Here, regardless of whether we place stiff or soft materials at each green pixel in the unit-cell, as long as the stiff and soft materials are in the positions defined by the template in yellow and purple, there will be a band gap within the user's desired range. **c**. The patterns are learned from coarse resolution training data but can be robustly transfered to finer resolution, and generate fine resolution unit-cells with the target property.

Metamaterials are traditionally designed through empirical trial-and-error or intuition [99] or computationally expensive topology optimization [99–103] which often produces designs out of predetermined geometrical building blocks. Recently, machine learning methods (ML) have gain popularity for metamaterial design. For example, machine learning models can be trained to predict material properties from unit cells defined by a finite set of pixels/voxels, and used as faster surrogate models for computationally expensive simulations [104–108]. Using deep generative models and neural network inversion techniques, some recent works [109–115] aim to directly solve the inverse design problem for metamaterials, i.e., generating the designs given the target property. However, the ML models used in these work are usually gigantic black boxes, whose decision processes are hard to understand. This is undesirable for scientific discovery purposes because scientists may also want to gain insights into what geometric features are important for a given target property, such a particular frequency band gap. In addition, these massive models are data hungry and not robust to distributional shift – they usually require a huge simulated dataset that covers most of the design space. These models also tend to perform poorly on datasets they have not seen before, such as unit-cells in a finer resolution space.

In this paper, instead of relying on existing black box approaches, we propose two novel rule-based ML approaches for metamaterial design that have major advantages:

- ***Interpretability:*** The approach allows us to discover interpretable key patterns within unit-cells that are related to a physical property of interest (see Figure 3.1**a** and 3.1**b**). We consider two types of patterns: (i) local patterns called *shape frequency features*, which calculate the occurrence frequency of certain shapes in the unit-cell; (ii) global patterns, called *unit-cell templates*, which look for arrangements of constituent materials in specific regions of the metamaterials' unit-cells. The unit-cell templates are optimized with binary integer programming to find global patterns within unit cells that give the metamaterial a desired property.

- ***Leverages Multi-resolution Properties:*** An important observation underpinning our methodology is that a pattern in the coarser resolution design space also exists in finer resolution design space, with one coarse pixel replaced by many finer pixels. As a result, *if a pattern can robustly characterize the target property at the coarse resolution design space, it will also be predictive at the finer resolution design space*. This leads to computationally-efficient discovery of many valuable metamaterial designs possessing the desired properties. In particular, our method allows us to construct a *scaffold of patterns* that allows interpretable coarse scale information discovered at low resolutions to be reliably transferred to make accurate predictions for high resolution designs (see Figure 3.1**c**).

- ***Flexible Metamaterial Designs:*** Our unit-cell templates (e.g., the one in Figure 3.1**b**) enables flexibility in unit-cell designs at any resolution. Simply, unit-cell templates specify regions where one can almost freely design unit-cell features without changing the target band gap property (e.g., in Fig 3.1**b** constituent phases can be arranged at will in the green regions determined by our algorithm). Such flexibility in design might be useful to satisfy practicality constraints such as connectivity or other design constraints such as overall stiffness.

Section 3.3 discusses related works on ML approaches for metamaterial designs. In Section 3.4, we introduce the problem setting. In Section 3.5, we provide the proposed methods and explain how they deal with the four core objectives of data-driven approaches that are important to materials scientists: (a) design-to-property prediction; (b) property-to-design sampling; (c) identify key patterns; (d) transfer to finer resolution. We then evaluate performance of the proposed methods in Section 3.6 and test them on practical applications. We discuss and conclude in Section 3.7.

3.3 Related Works

Metamaterials are architected materials with engineered geometrical micro- and meso-structures that can lead to uncommon physical properties. As mentioned in the introduction, many existing works apply machine learning models for designing metamaterials, i.e., assembling constituent materials into metamaterials that have specific physical properties.

Much past work focuses on structure-to-property prediction. Because of the expensive computational cost of numerical simulations, these works train machine learning models (mostly deep learning models) to approximate the simulation results [104–108]. Using these ML models as a fast replacement of the simulator, materials satisfying the design objective can be found more efficiently using rejection sampling, i.e., randomly picking a structure in the design space until it satisfies the design objective. However, given the immense size of the design space, finding materials through rejection sampling can still be computationally inefficient. Therefore, some recent works apply deep generative models and neural network inverse modeling [109–115] to train a more efficient materials sampler, aiming to solve the inverse design problems for metamaterials, i.e., property-to-structure sampling. See [116–118] for reviews on using deep learning methods for metamaterial designs.

Almost all modern existing work on this topic uses black box models like deep neural networks. Such approaches are unable to answer key questions such as “What patterns in a material’s design would lead to a specific desirable property?” A link between the specific design and the target property could be useful for further research; i.e., to determine whether there is an agreement with domain knowledge, and if not, to potentially discover new knowledge.

Other work also stresses the importance of interpretability for metamaterial design [93, 112, 119]. These works are very different from ours, and cannot solve the challenge we want to address. Ma et al. [112] try to make the latent space of deep generative models interpretable. Their interpretable features describe general geometrical information such as size and shape of holes in the material but these properties are not associated with the target property. In contrast, our goal is to find key patterns within unit cells that result in the target property (in our case, a band gap). Elzouka et al. [119] and Zhu et al. [93] also build rule-based models, namely decision trees, but the design problems they are solving are different and much simpler than the problem we try to solve. Specifically, their materials are described by several continuous features, e.g., thickness of the materials, while we work on pixelated metamaterials whose features are just raw pixels made of constituent materials. Discovering interpretable patterns directly from raw pixels is a more challenging and fundamental problem.

Another serious issue with using black box models like deep neural networks is that they require large labeled training datasets, and the training and testing data should come from the same distribution so that the model generalizes between training and test. However, constructing these large datasets is extremely expensive because the labels (i.e., material properties) of metamaterials are calculated by simulation, which is computationally expensive. In fact, this whole process could be so expensive that one might find it less expensive to use the simulator to get the results on the test set directly rather than go through the process of collecting a training set at all. Note that deep generative models like GANs [120, 121] do not address the simulation bottleneck because training a GAN requires co-training a generator and a discriminator which requires even more training data than just training a structure-to-property predictor. Ma et al. [122] propose a self-supervised learning approach that can utilize randomly generated unlabeled data during training to reduce the amount of training data. However, this work can only handle structure-to-property prediction but not the problem of inverse design considered here. Our method instead alleviates this simulation bottleneck through a *multi-resolution* approach: our unit-cell template models are trained using a (relatively small amount of) *coarse-resolution* data but can extrapolate and generate a (large amount of) *finer-resolution* metamaterial designs. This is helpful because the coarse resolution space is much smaller than the fine-resolution space, and gives us a bird's eye view of what might happen when we sample at the finer scale throughout the space of possible metamaterials.

3.4 Problem Setting

Here, we introduce the setting of the metamaterial design problem we are trying to solve, including the inputs and target of the dataset and their physical meanings.

We aim to design and characterize 2-D pixelated metamaterials made by tiling a 10×10 unit-

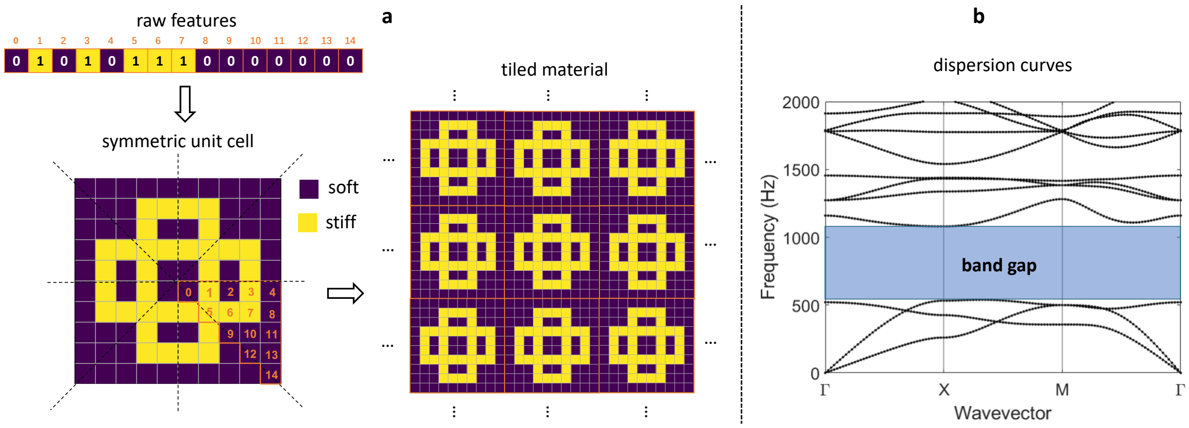


Figure 3.2: Input and target of our metamaterial design problem. **a.** 2-D phononic material constructed with the raw features. Upper Left: raw 15 dimensional input feature vector. Lower left: the feature vector defines the triangle in the lower right of the unit-cell. The triangle is copied using lines of symmetry to define the full unit-cell. Right: the unit-cell is tiled to obtain the full material. For the Bloch-Floquet boundary conditions, the tiling is infinite in all dimensions; **b.** The design objective is the frequency band gaps.

cell. Such materials can be stacked to form 3D structures, and can direct, reflect or scatter waves, depending on the choice of unit-cell's material selection and geometry. In our framework, the unit-cell is a square with side length $a = 0.1$ m. However, transferring to a different length scale for a different application is easily doable by a simple scaling transformation on the dispersion relations. For a 10×10 unit-cell, the pixel side length is 1 cm; for 20×20 unit-cell, the pixel side length is 0.5 cm. Each unit-cell is made of two constituent materials: one is soft and lightweight, with elastic modulus $E = 2$ GPa[1] and density $\rho = 1{,}000$ kg/m^3, and the other is stiff and heavy with $E = 200$ GPa, and $\rho = 8{,}000$ kg/m^3. These two sets of material properties are representative of a polymer and steel respectively. Our unit-cells are symmetric, with four axes of symmetry (x,y and ±45°). Under the symmetry constraints, the coarsest resolution (10×10) unit-cell has only 15 irreducible pixels. As a result, the raw input features of a sample in our dataset is a 15-dimensional binary vector: 0 means the soft constituent material in that location, and 1 means stiff constituent material. Thus, the full coarse space can be characterized, having 2^{15} total states. Figure 3.2 shows how to construct a material from the representation involving the 15 raw input features.

The material property we desire in our engineered materials is a band gap within a specific frequency range, given by the user. A band gap is a range of frequencies within which elastic waves cannot propagate and are instead reflected.

To identify the existence of a band gap, one can examine the effect of dispersion in metamaterials by calculating dispersion relations. Dispersion relations are functions that relate the wavenumber of a

---

[1] GPa is gigapascals, a unit used to quantify elastic modulus.

wave to its frequency, and they contain information regarding the frequency dependent propagation and attenuation of waves. Dispersion relations are found by computing elastic wave propagation solutions over a dense grid of wavevectors. A band gap exists when there is a range of frequencies in the dispersion relation for which no wave propagation solutions exist (see Figure 3.2 **b**).

Dispersion relation computations use Bloch-Floquet periodic boundary conditions, i.e., they assume that a given unit-cell is tiled infinitely in space. The physics revealed in dispersion analysis (infinite-tiling) can be leveraged in more realistic finite-tiling scenarios, as we will demonstrate later, which makes dispersion relation computations very useful for exploration and design of real materials. Our dispersion relation simulations are implemented using the finite element method.

More details about the simulation can be found in the Supplementary Information A.

We are looking for materials with band gaps in a certain frequency range. To define this task as a supervised classification problem, we create a binary label based on existence of a band gap in a given frequency range (e.g. [10, 20] kHz): 1 means one or more band gaps exist, 0 means no band gap exists. In other words, if a band gap range intersects with the target frequency range, the label is 1, otherwise the label is 0. One can also flexibly adjust the band gap label for different practical uses. For example, we can set the label to 1 only when the intersection of the band gap and the target frequency range is above a minimum threshold. We can also set the label to 1 when the band gap covers the entire target range, or even create a label for band gap properties in multiple frequency ranges.

3.5 Method

This section introduces proposed methods. Section 3.5.1 explains the shape-frequency features and how they can be used to optimize different objectives. Inspired by the efficiency of shape-frequency features, we then propose unit-cell template sets in Section 3.5.2.

3.5.1 Shape-Frequency Features

We hypothesize that the occurrence of certain local features in the metamaterials might contribute to the formation of band gaps. For example, certain local patterns/shapes in the materials can lead to interference and thus cause band gaps. Because the unit-cells are repeated, the location of such local patterns does not matter, *as long as they occur frequently*, the band gap can be formed. Such physics intuition inspires us to propose *shape-frequency features* which calculates the *number of times a pattern occurs in the unit cell divided by total number of locations*.

Denote the unit-cell as a $n \times n$ binary matrix $\mathbf{U} \in \{0, 1\}^{n \times n}$, where $\mathbf{U}_{i,j} = 0$ means pixel i, j is assigned to the soft material and $\mathbf{U}_{i,j} = 1$ when the pixel is assigned to the stiff material.

A specific shape s can be represented as a set of location offsets O_s whose elements are coordinates of pixels with respect to a reference pixel. For example, a 2×2 square window can be represented as $\{(0,0),(1,0),(0,1),(1,1)\}$. A 3×3 plus symbol as in Figure 3.1 can be represented as $\{(0,1),(1,0),(1,1),(1,2),(2,1)\}$. For a specific unit-cell, the feature value corresponding to that shape is computed by sliding the shape over the unit-cell and calculating the fraction of times that it is entirely contained within the soft material,

$$f_s = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \mathbb{1}\left[\left(\sum_{(o_r,o_c)\in O_s} \mathbf{U}_{i+o_r,j+o_c} \right) = 0\right], \tag{3.1}$$

where $\mathbb{1}[\cdot]$ is the indicator function. It equals 1 if and only if all the pixels in the shape are soft material (i.e., $\sum_{o_r,o_c} \mathbf{U}_{i+o_r,j+o_c} = 0$).

We would typically consider a collection of shapes, and have one element in a unit-cell's feature vector per shape. Thus, for unit-cell i, the jth component of its feature vector corresponds to how often the full shape j appears in its soft material.

In more detail, consider the collection of shapes, shown within Figure 3.3**a** (left). Note that this collection of shapes is the full set used in the results section, not just examples of them. Again, pixels of the stiff material are in yellow, and the soft material is shown in purple. We slide each of the shapes (sliding windows) over the unit-cell (Figure 3.3**a** (middle)) and count the fraction of positions over which the shape is fully contained within pixels of the soft (purple) material. These fractions together form the new representation for the unit-cell (Figure 3.3**a** (right)). Note that, to calculate the fraction, we should also consider the situation where the sliding window is across the boundary of two unit-cells, since the entire material is made by tiling the unit-cell. Because the unit-cells are symmetric, the occurrence of the patterns within the unit-cell are also symmetric: if we rotated the patterns by 90°, 180° or 270°, the number of detections of the pattern within the unit-cell would be identical. Note that the shape-frequency features are different from standard convolution filters used in computer vision; details are discussed in the Supplementary Information B. Theoretically, our method can be generalized to nonsquare pixels and unit-cells as well, see Supplementary Information E.

Once we calculate the shape-frequency features of the unit-cells, they can replace the original raw features and be used as the inputs of the machine learning models to predict the band gap output. We show later in Section 3.6.1 that using the shape-frequency features as inputs, machine learning models can predict the existence of band gaps more accurately than using raw features.

Since shape-frequency features are just new representations of the unit-cells and they are written in vector form, any type of machine learning model can be trained to predict band gap existence, taking these features as inputs. These machine learning models can not only be complex models like neural networks or boosted trees, but can also be interpretable models (e.g., sparse decision

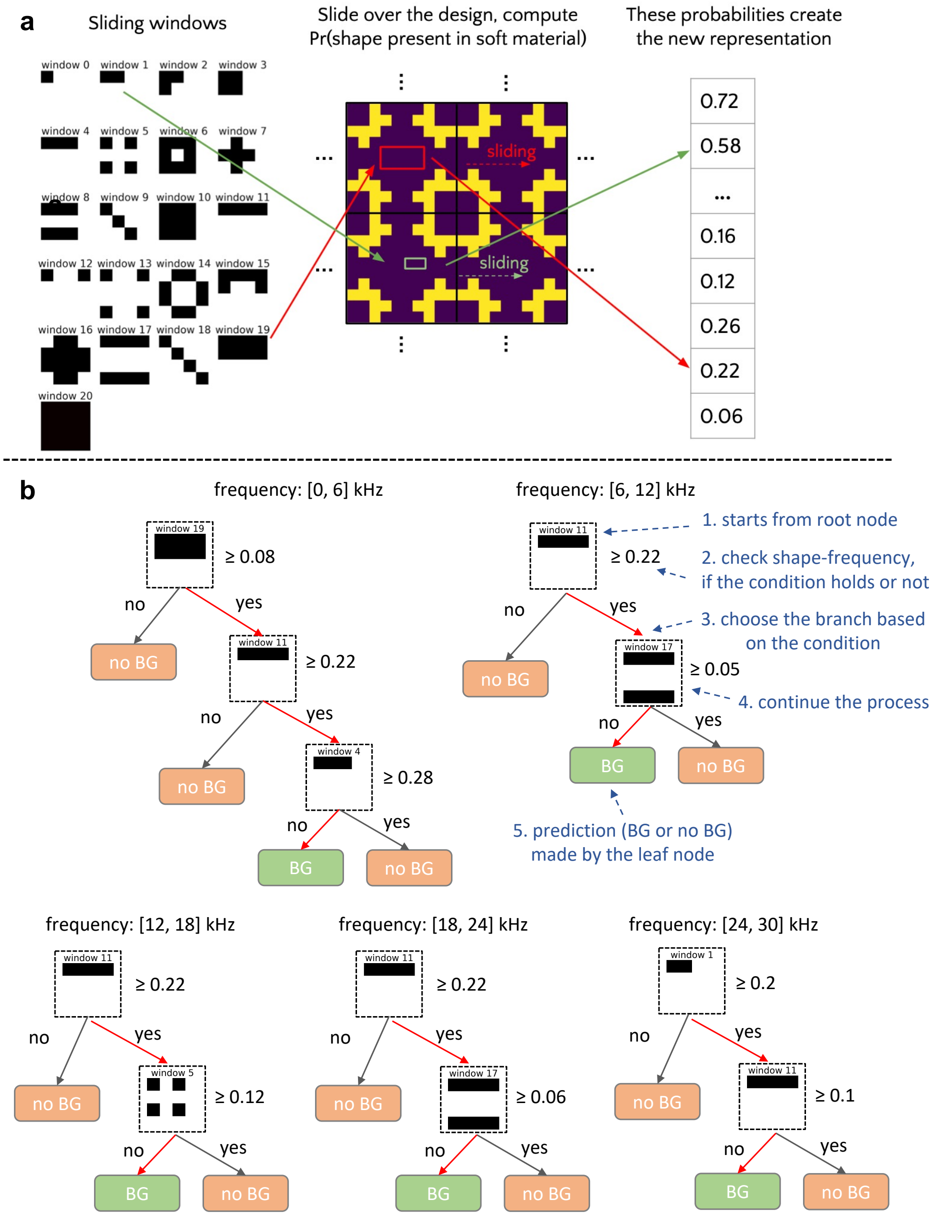


Figure 3.3: **a.** The process of calculating shape-frequency features. *Left*: Collection of shapes (sliding windows) used to create the shape-frequency features. *Middle:* An example 10 × 10 unit-cell design (tiled 4 times for better visualization). *Right:* Shape-frequency features of the unit-cell, which count the fraction of locations in the unit-cell where the shape is present in the soft material; **b.** Optimal sparse decision trees built on shape-frequency features for predicting band gaps in different frequency ranges; red arrows denote the paths to band gap (BG) nodes. Blue text on the top-right tree breaks down how the decision tree predicts whether the band gap exists using shape-frequency features.

trees). Figure 3.3**b** shows examples of sparse decision trees that predict the existence of band gaps from shape frequency features. When making predictions, the decision tree starts from its root node, checking if the shape shown on the node appears frequently in the unit-cell (e.g., if a 1×4 soft bar occurs in the unit-cell more than 22% of all possible locations). If so, it goes to the right branch; if not it goes to the left branch. The process is continued until a leaf node (in green or orange) is reached. At that point, it outputs the prediction of whether a band gap exists, based on the majority vote of the training data within that leaf. The paths denoted by red arrows in the trees in Figure 3.3**b** show how often the local patterns need to occur in the unit-cells to predict an open band gap. More analysis of these discovered patterns can be found in Section 3.6.3.

3.5.1.1 Optimizing Precision and Support

For regular binary classification problems, we hope the machine learning models have high accuracy for both positive samples (materials with band gaps) and negative samples (materials with no band gap). This is also the objective if our goal is to perform only structure-to-property prediction. However, for the property-to-structure task, where our goal is to produce a number of unit-cell designs with the target band gap property, prediction accuracy is no longer a good metric. Instead, we hope all unit-cells that are predicted to have a band gap actually have a band gap, i.e., we prefer that the model has high *precision*. We also hope the total number of discovered designs, i.e., the *support*, meets the requirement of real applications. Thus, for the property-to-structure task, our objective is a combination of precision and support.

Most machine learning methods cannot directly optimize custom objectives with constraints, such as precision, constrained by support. However, there are new approaches that permit direct optimization of custom discrete objectives. We use GOSDT (Generalized and Scalable Optimal Sparse Decision Trees, [123]) for this task, because it directly optimizes decision trees for customized objectives. Different from traditional decision tree algorithms which use greedy splitting and pruning, GOSDT directly searches through the space of all possible tree structures, uses analytical bounds to reduce a huge amount of search space, and directly outputs a tree that optimizes the customized objective. We programmed it to maximize the following custom objective to optimality:

$$\max_{\text{tree}} \left[\frac{TP_{\text{tree}}}{TP_{\text{tree}} + FP_{\text{tree}} + \epsilon} - \frac{K}{TP_{\text{tree}} + \epsilon} \right] \tag{3.2}$$

$$= \max\nolimits_{\text{tree}} \left[\frac{P - FN_{\text{tree}}}{P - FN_{\text{tree}} + FP_{\text{tree}} + \epsilon} - \frac{K}{P - FN_{\text{tree}} + \epsilon} \right]. \tag{3.3}$$

Here, K is a parameter that balances the precision and the support; ϵ is a small constant for numerical convenience; TP, FP, TN, FN mean true positives, false postives, true negatives and false negatives. Equation (3.2) shows precision (first term) and inverse support (second term); we use inverse support so that if support is large, the term diminishes in importance. The simplification in Equation (3.3) shows that the objective is monotonically decreasing with respect to FN_{tree} and FP_{tree}. Theorem B.1 of [123] shows that as long as the objective is decreasing with respect to FN_{tree} and FP_{tree}, we can find an optimal sparse decision tree using GOSDT's branch and bound algorithm.

3.5.1.2 Property-to-structure Sampling

Because we optimize the precision using GOSDT, the false positive rate of our model will be low enough to work with. At this point, we directly do rejection sampling using the decision tree to produce unit-cell designs with the target band gap. Specifically, to produce valid designs, the rejection sampling approach randomly picks structures in the design space, evaluates whether each of them are predicted to have a band gap, and outputs only these relevant designs. After sampling, each accepted sample is evaluated with the physics-based finite-element model to determine whether a band gap is present.

3.5.1.3 Transfer to Finer Resolution

Algorithm 3.5.1 Sampling Fine resolution Unit-cell Designs via Shape-frequency Features

```
Input: simulated coarse resolution (10 × 10) dataset $\mathcal{D} := \{\mathbf{x}_i, y_i\}_{i=1}^{2^{15}}$, $\mathbf{x}_i \in \{0,1\}^{15}$: raw features; $y_i$: band gap la
Parameters: set of shapes $S$; tree sparsity regularization $\lambda$ (see [123]); $K$, $\epsilon$ (see Section 3.5.1.1)
Output: raw features of a fine resolution unit-cell $\tilde{\mathbf{x}}$
calculate shape-frequency features $\mathbf{x}_i^{\text{SFF}} = \text{SFF}(\mathbf{x}_i, S)$ for all $\mathbf{x}_i$ in the dataset (coarse resolution), see Section 3
train an optimal sparse decision tree $\tau = \text{GOSDT}(\{\mathbf{x}_i^{\text{SFF}}, y_i\}_{i=1}^{2^{15}}, \lambda, K, \epsilon)$, see Section 3.5.1.2
while(True):
    randomly sample a binary vector $\tilde{\mathbf{x}}$ as raw features in the fine resolution space
    calculate shape-frequency features $\tilde{\mathbf{x}}^{\text{SFF}} = \text{SFF}(\tilde{\mathbf{x}}, S)$ (fine resolution), see Section 3.5.1.3
    if $\tau(\tilde{\mathbf{x}}^{\text{SFF}}) = 1$:
        return $\tilde{\mathbf{x}}$
```

The raw feature space of finer resolution samples is different from that of coarse resolution samples. Therefore, for the finer resolution data, we need to slightly modify the approach to obtain shape-frequency features that are compatible with the coarse resolution shape-frequency features. Suppose we want to transfer the model from 10 × 10 to 20 × 20 space. Then, there are three changes in calculating the shape-frequency features:

- The window size should be doubled when moving from coarse resolution to fine resolution;
- The stride of the sliding window should be 2 instead of 1;

- In counting the shape-frequency values, exact agreement between the window and soft material (purple) should be replaced with near exact agreement. In particular, when less than 2 yellow pixels (stiff material) are found in the window, we can consider this to be an agreement.

When using the shape-frequency features with these modifications, the decision tree model learned on the coarse resolution data can be directly applied to the fine resolution. Thus, we can also do rejection sampling on the fine resolution with the model.

Algorithm 3.5.1 shows the entire pipeline of using an optimal sparse decision tree built on shape-frequency features to sample fine resolution designs with the target band gap property. Visual illustration of the sampling process can be found in Supplementary Information C.

3.5.2 Unit-cell Template Sets

Here, we introduce another interpretable machine learning model, called unit-cell template sets. Different from sparse trees on shape-frequency features, which focus on local patterns, a unit-cell template captures a global pattern for the unit-cell that is related to the target properties. The unit-cell template is a $n \times n$ matrix $\mathbf{T} \in \{0, 1, *\}^{n\times n}$, where $\mathbf{T}_{i,j} = 0$ means the pixel is soft material, $\mathbf{T}_{i,j} = 1$ means the pixel is stiff material, and $\mathbf{T}_{i,j} = *$ means the pixel could be either soft or stiff, i.e., a free pixel.

Definition *(match)*. We say a unit-cell design $\mathbf{U}$ *matches* the unit-cell template if and only if all pixels with value 0 on the template are also 0 on the design, and all pixels with value 1 on the unit-cell template are also 1 on the design. That is, $\forall(i, j)$ such that $\mathbf{T}_{i,j} \neq *$, we have $\mathbf{U}_{i,j} = \mathbf{T}_{i,j}$.

A unit-cell template set contains a set of unit-cell templates, and the sample design is predicted as positive if and only if it matches at least one unit-cell template in the set. Figure 3.4**a** shows an example of a unit-cell template set that consists of five different templates. We proposed unit-cell templates because we found that some pixels in the unit-cells are more important for the formation of band gaps than others. For the pixels that are not important, even if they are flipped, the band gaps remain unchanged; we denote these as free pixels in the unit-cell template. The free pixels identify unimportant regions where changes do not affect the target band gaps, while the other pixels form the key global pattern that leads to the band gap. We aim to find a unit-cell template set that captures a diverse set of global patterns related to the target band gap. The relationships between unit-cell template sets and other machine learning methods are discussed in Supplementary Information B. Theoretically, our method can be generalized to nonsquare pixels and unit-cells as well, see Supplementary Information E.

The training objective of the model is to find a small number of unit-cell templates, such that the

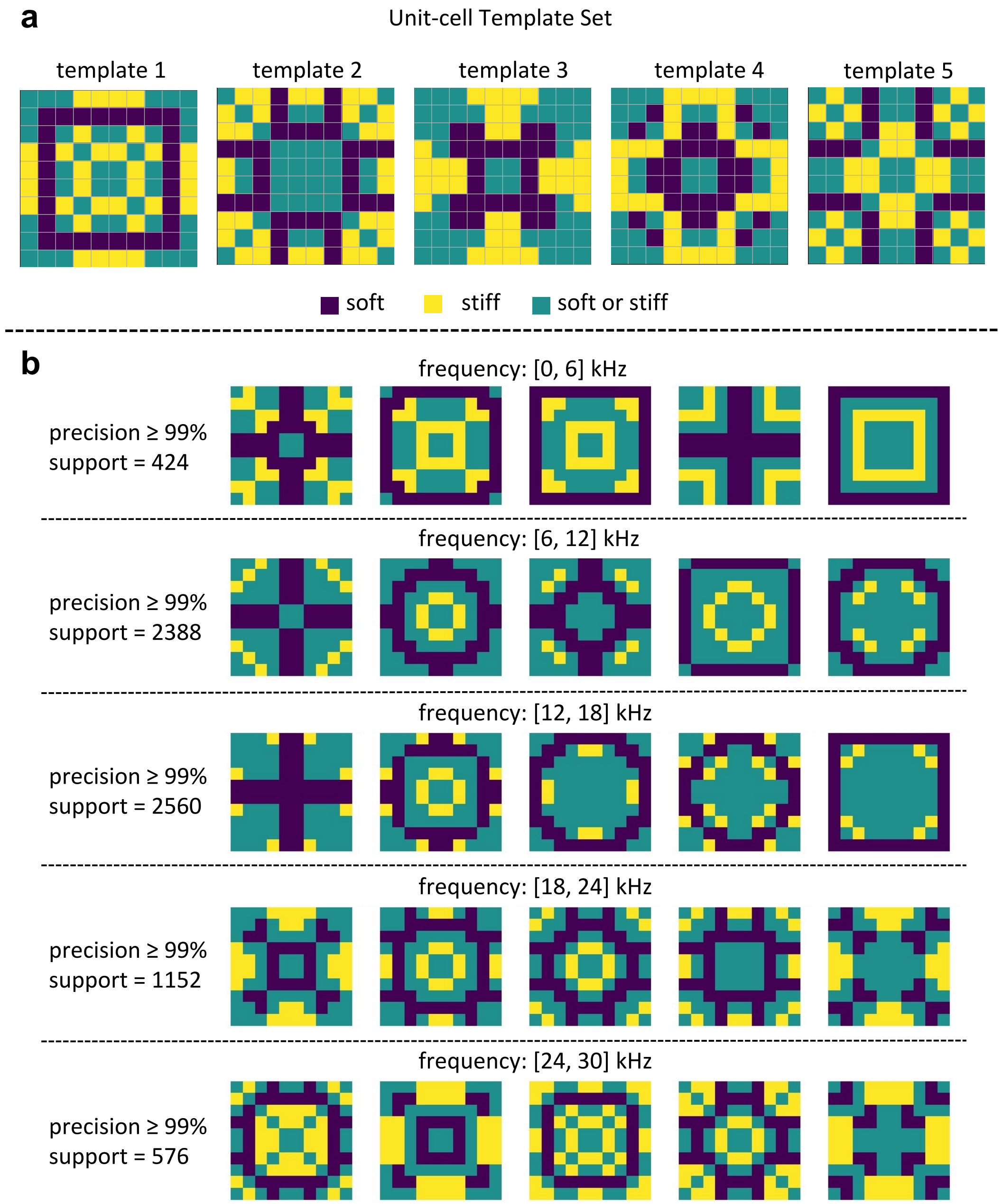


Figure 3.4: **a**. An Example of a unit-cell template set. A unit-cell is predicted as positive as long as it matches at least one unit-cell template in the set; **b**. unit-cell template sets learned for predicting band gaps in different frequency ranges. Note that because unit-cells are tiled, a large cross through the center is identical to a square on the border. (E.g., consider the two upper right unit-cells.)

training precision of the entire model is high enough, and the model covers as many valid designs as possible, i.e., maximizing the support under a minimum precision constraint. The reasons for optimizing precision and support were explained in Section 3.5.1.1, and we set a limit for the total number of selected unit-cell templates to encourage the unit-cell template sets to contain a diverse set of unit-cell templates. Because the total number of possible templates is extremely large, the training process is divided into two steps, a pre-selection of candidate unit-cell templates that filters out useless templates to reduce the problem size (Section 3.5.2.1), and a integer linear programming (ILP) formulation to optimally select from the candidates obtained in the first step (Section 3.5.2.2). Figure 3.4**b** shows the unit-cell template sets learned by the proposed algorithm for band gap prediction. More analysis of these discovered patterns can be found in Section 3.6.3.

3.5.2.1 Pre-selection of Templates

Here we consider symmetric unit-cell templates, because designs in our dataset are all symmetric. By removing all symmetry redundancy, the unit-cell template can be represented by its irreducible pixels, i.e. a 15 dimensional vector $\mathbf{t} \in \{0, 1, *\}^{15}$. The total number of possible templates is 3^{15} which is approximately 14.3 million. This is too large for the ILP in the next step. However, among the 3^{15} possible templates, most of them would never be selected because either their precision or support is not high enough. For example, a unit-cell template with precision 80% is not likely to be used if we want the entire model to have precision above 99%, i.e., we hope the unit-cells are all having the desired band gap properties. Also, if the support of a unit-cell template, i.e., the total number of designs that match it, is very small (e.g., < 10), the model may not generalize well. Therefore, we pre-select the unit-cell templates by setting minimum thresholds of precision and support, which reduces the search space only to promising unit-cell templates. We select the unit-cell template as a candidate only when it meets the minimum thresholds.

Observing that the entries are all binary in the unit-cell designs, we implement the precision calculation via bit operations, which significantly improves the speed of the pre-selection step. With the bit-operation implementation, the pre-selection steps of all 3^{15} possible templates finish in 50 seconds. The number of unit-cell templates that remains is typically in the range of 6000 to 12000, which is now suitable for ILP.

3.5.2.2 ILP for Template Selection

After the pre-selection step, we have a set of candidate unit-cell templates. Since the pre-selection step significantly cuts down the space of templates, we can directly formulate the template selection as an optimization problem, and solve it to provable optimality. Specifically, we formulate a ILP to optimally select from the candidates. Suppose we have n designs and m candidate templates. The goal of the ILP is to choose at most s unit-cell templates ($s \ll m$) whose union forms a model, such that the support is maximized and the precision of the model is at least p. Denote the true labels of all designs by a binary vector $\mathbf{y} \in \{0,1\}^n$, and the predicted labels by $\hat{\mathbf{y}} \in \{0,1\}^n$. $\mathbf{M} \in \{0,1\}^{n\times m}$ denotes a matching matrix, where $M_{i,j}$ indicates whether design i matches template j (1 for match, and 0 for not match). Binary vector $\mathbf{c} \in \{0,1\}^m$ denotes the chosen unit-cell templates, where c_j indicates whether template j is chosen (1 for choose, and 0 for not choose). We solve the following ILP for template selection.

$$\max \sum_{i=1}^{n} \hat{y}_i \quad \text{(optimizing support)} \tag{3.4}$$

$$\text{s.t.} \sum_{j=1}^{m} c_j \le s \quad \text{(sparsity constraint)} \tag{3.5}$$

$$\sum_{i=1}^{n} y_i \cdot \hat{y}_i \ge \left(\sum_{i=1}^{n} \hat{y}_i\right) \cdot p \quad \text{(minimum precision)} \tag{3.6}$$

$$\sum_{j=1}^{m} M_{i,j} \cdot c_j \ge \hat{y}_i, \;\; i = 1, \ldots, n \quad \text{(define } \hat{y}_i\text{)} \tag{3.7}$$

$$\sum_{j=1}^{m} M_{i,j} \cdot c_j \le m \cdot \hat{y}_i, \;\; i = 1, \ldots, n \quad \text{(define } \hat{y}_i\text{)} \tag{3.8}$$

$$c_j \in \{0,1\}, \quad j = 1, \ldots, m \tag{3.9}$$

$$\hat{y}_i \in \{0,1\}, \quad i = 1, \ldots, n. \tag{3.10}$$

In this ILP, the objective (3.4) means maximizing the total number of designs predicted as positive, which is the same as support. Constraint (3.5) controls the sparsity, i.e., choose at most s unit-cell templates. Constraint (3.6) guarantees training precision of the model is at least p. (3.7) and (3.8) constraints together define $\hat{y}_i$, where $\hat{y}_i = 1$ if and only if design i matches at least one of the chosen unit-cell templates. In particular, (3.7) says that if design i does not match any chosen template j (i.e., whenever c_j is 1, $M_{i,j}$ happens to be 0), then $\hat{y}_i$ will be set to 0. (3.8) will ensure that if there is a match for design i to any of the chosen templates (which all have $c_j = 1$), then this design is assigned $\hat{y}_i$=1. Using a commercial MIP solver, a problem with around 10000 candidate templates can be solved to optimality (when the current best solution meets the upper bound of the best possible solution) or near-optimality in about 10 minutes (running single-threaded on one core of a 2.66GHz Intel E5640 Xeon Processor). If $s = 5$, it can be solved to optimality all the time, and when $s = 10$, the optimality gap (difference between current best solution and an upper bound of the best possible solution as the percentage of the upper bound) is always <20% for a run time of 30 minutes (running under the same environment); it is worthwhile to note that optimal solutions are often attained quickly, but the solvers can take a while to prove that the solution is optimal. Note that, if desired, one can set higher minimum precision and support thresholds for the pre-selection step to make the problem even smaller, so that the ILP can be solved even faster. We choose $s = 5$ for all the experiments in the main paper.

The result of the ILP is our unit-cell template set.

3.5.2.3 Property-to-structure Sampling

After training the structure-to-property model, the resulting unit-cell template set can be directly used to solve the inverse property-to-structure problem. An easy sampling procedure to do this is as follows: first, randomly choose a unit-cell template **t** from the unit-cell template set, where the probability to choose each template is proportional to its support; second, for all entries in **t** that equal *, randomly assign value 0 or 1 to them. These sampled unit-cells are likely to have the desired band gap.

3.5.2.4 Transfer to Finer Resolution

The unit-cell template set naturally transfers coarse scale information to finer resolutions. In particular, by subdividing each pixel in the unit-cell template into four sub-pixels, we directly obtain a unit-cell template defined on a finer-resolution space.

Algorithm 3.5.2 shows the entire pipeline of using unit-cell template set to sample fine resolution designs with the target band gap property. Visual illustration of the sampling process can be found in Supplementary Information C.

Algorithm 3.5.2 Sampling Fine Resolution Unit-cell Designs via Unit-cell Template Set

```
Input: simulated coarse resolution (10 × 10) dataset $\mathcal{D} := \{\mathbf{x}_i, y_i\}_{i=1}^{2^{15}}$, $\mathbf{x}_i \in \{0,1\}^{15}$: raw features; $y_i$: band gap la
Parameters: pre-selection support $\psi_{\text{pre}}$, pre-selection precision $p_{\text{pre}}$, sparsity constraint $s$, minimum precision $p$
Output: raw features of a fine resolution unit-cell $\tilde{\mathbf{x}}$
pre-select candidate template set $S_T^{\text{pre}}$ = pre-selecting($\mathcal{D}$, $\{0, 1, *\}^{15}$, $\psi_{\text{pre}}$, $p_{\text{pre}}$), see Section 3.5.2.1
run ILP to find optimal template set $S_T^*$ = ILP($\mathcal{D}$, $S_T^{\text{pre}}$, $s$, $p$), see Section 3.5.2.2
randomly pick a template $\mathbf{t} \in S_T^*$
expand $\mathbf{t}$ to fine resolution space, get $\tilde{\mathbf{t}}$
randomly set to 0 or 1 for all * elements in $\tilde{\mathbf{t}}$, get $\tilde{\mathbf{x}}$
return $\tilde{\mathbf{x}}$
```

3.6 Results

The results are organized according to four objectives we want to achieve with the proposed methods, including (a) design-to-property prediction; (b) property-to design sampling; (c) identify key patterns; (d) transfer to finer resolution. We evaluate how well the proposed methods can achieve these objectives, followed by several tests involving practical applications in materials discovery.

3.6.1 Objective 1: Structure-to-property Prediction

Here, we test how well the proposed methods can perform structure-to-property prediction. We chose five frequency ranges ([0, 10], [10, 20], [20, 30], [30, 40] and [40, 50] kHz) to predict the existence of band gaps; these frequency ranges correspond to five different binary classification problems. Using balanced accuracy (bacc) as the evaluation metric, we compare the predictive performance of a diverse set of ML models with and without the shape-frequency features. We specifically consider linear models like support vector machines with linear kernels (SVMs) and logistic regression (LR); tree-based models like CART, random forest (RF), and boosted trees (LightGBM [124]), as well as neural networks including the multi-layer perceptron (MLP). We also compare the proposed method with convolutional neural networks (CNNs), since they have been widely used in previous works of ML-based metamaterial design. The baccs of each model trained on raw feature, SFF, and improvements of SFF over raw features, are shown in Table 3.1 (a). We train each model 5 times and average the accuracy.

Our results show that **using the shape-frequency features, rather than the original raw features, improves the accuracy of classifiers for most machine learning methods**, especially tree-based methods such as boosted trees, but with the exception of MLP (SFF decreases its in [0, 10] kHz, [10, 20] kHz, and [20, 30] kHz).

One might expect CNNs to achieve great success in classifying band gaps for 2-D metamaterials since the unit-cells share many similarities with images. However, LightGBM [124] built on shape-frequency features outperforms ResNet18 [125] in all ranges. In some cases, e.g., within frequency range [40, 50] kHz, simple models like CART outperform CNNs.

More details of the experiment (e.g., hyper-parameter settings) can be found in Supplementary Information D.1.

3.6.2 Objective 2: Property-to-structure Sampling

Using the methods discussed in Section 3.5.1.2 and Section 3.5.2.3, we are able to solve the inverse design (property-to-structure sampling) problem.

In practice, materials scientists need valid designs with the target property, but they do not require

Model		Frequency range				
		[0, 10]kHz	[10, 20]kHz	[20, 30]kHz	[30, 40]kHz	[40, 50]kHz
SVM	raw	71.77%	73.88%	50.85%	54.93%	49.84%
	SFF (Ours)	75.62%	77.35%	67.96%	59.89%	49.93%
	improvement	+3.85%	+3.47%	+17.11%	+4.96%	0.09%
LR	raw	78.03%	75.44%	56.31%	76.59%	90.96%
	SFF (Ours)	80.53%	79.55%	69.04%	76.9%	91.32%
	improvement	+2.50%	+4.11%	+12.73%	+0.31%	+0.36%
RF	raw	85.81%	80.04%	73.00%	77.66%	87.54%
	SFF (Ours)	85.52%	81.98%	74.53%	81.26%	**95.35%**
	improvement	-0.29%	+1.94%	+1.53%	+3.60%	+7.81%
CART	raw	84.74%	75.68%	63.40%	73.95%	86.82%
	SFF (Ours)	83.63%	80.13%	70.72%	79.73%	94.47%
	improvement	-1.11%	+4.45%	+7.32%	+5.78%	+7.65%
MLP	raw	90.72%	86.39%	78.15%	77.47%	65.90%
	SFF (Ours)	85.90%	82.55%	76.01%	77.69%	66.4%
	improvement	-4.82%	-3.84%	-2.14%	+0.22%	+0.50%
LightGBM	raw	91.32%	88.27%	81.11%	82.62%	77.76%
	SFF (Ours)	**96.10%**	**90.97%**	**85.57%**	**90.01%**	94.76%
	improvement	+4.78%	+2.70%	+4.46%	+7.39%	+17.00%
CNN	raw	93.24%	89.76%	81.65%	79.56%	84.83%

(a) structure-to-property prediction: testing balanced accuracies (baccs) of different methods. We mark the models with the best bacc in each frequency range in bold.

Frequency range	Raw+LightGBM	SFF+LightGBM	SFF+GOSDT (Ours)	Unit-cell Template Sets (Ours)
[0, 10]kHz	80.77%, 2310	88.93%, 2169	95.77%, 89	**98.53%**, 339
[10, 20]kHz	93.52%, 4013	95.62%, 4063	**98.11%**, 423	**98.68%**, 758
[20, 30]kHz	86.94%, 3654	89.56%, 3811	**94.15%**, 205	**94.08%**, 203

(b) Property-to-structure sampling: testing precision and support of different methods. Numbers in the table cells are formatted as "precision, support." The testing support here is calculated among 6554 testing samples (20% of the entire dataset).

Frequency range	CNN+resizing	LightGBM+resizing	SFF+GOSDT (Ours)	Unit-cell Template Sets (Ours)		
	20 × 20	20 × 20	20 × 20	20 × 20	40 × 40	80 × 80
[0, 10]kHz	18.0%	25.0%	72.5%	**100.0%**	**100.0%**	**100.0%**
[10, 20]kHz	58.0%	68.5%	73.5%	**98.5%**	**99.0%**	**100.0%**
[20, 30]kHz	39.5%	52.5%	25.0%	**91.0%**	**96.0%**	**98.0%**

(c) Transfer to finer resolution: transfer precision of different methods.

Table 3.1: Summary of key quantitative results. The results are organized with respect to different objectives of data-driven metamaterials design. (a) Structure-to-property prediction; (b) Property-to-structure sampling; (c) Transfer to finer resolution.

the set of *all* designs with the property. As such, our performance metric is precision, rather than recall. We also calculate the support, which is the total number of testing samples predicted as positive, to ensure the models can generate enough potentially-valid designs. Table 3.1 (b) lists the precision and support values from different methods. The methods we compared include GOSDT trained on shape-frequency features (denoted SFF) with the objective in Section 3.5.1.1, the unit-cell template set, and LightGBMs trained on SFF and raw features.

In terms of precision, **SFF+GOSDT and unit-cell template sets significantly outperformed LightGBMs**. This is probably owing to the fact that the proposed methods directly optimize precision. LightGBMs maintain larger support, while the support of SFF+GOSDT and unit-cell template sets is much lower. But for practical use, it is sufficient that the model finds dozens of valid designs. The average sampling time of these methods and the results of unit-cell template sets with different sparsity constraints can be found in Supplementary Information D.2.

3.6.3 Objective 3: Show Key Patterns

One advantage of the proposed methods is model interpretability; we aim to explicitly identify the key patterns learned from data that are related to the target property. In this way, domain experts can verify whether the learned rules are aligned with the domain knowledge, or even discover new knowledge.

In Figure 3.3**b** and Figure 3.4**b**, we visualize the GOSDT+SFF and unit-cell template set learned for band gaps in several frequency ranges ([0, 6], [6, 12], [12, 18], [18, 24], [24, 30] kHz).

In Figure 3.3**b**, the top splits of each tree trained on shape-frequency features seem to be looking for bars in the soft material. Taking the top-right tree in 3.3**b** as an example, the root node checks if the 1×4 soft bar occurs in more than 22% of the places in the unit-cell. The unit-cell frequencies need to pass the thresholds to get to the "band gap" node. All band gap nodes are on the left branch of the deepest decision nodes in the trees. For instance, the deepest decision node of the top-right tree checks if shape 17, two 1×4 soft bars 2 pixels away from each other, occurs with more than 5% frequency in the unit-cell. To reach the band gap prediction node, the sample needs to go to the left branch, where shape 17 should occur with less than 5% frequency. This indicates the unit-cells should not have too many soft material patterns. In the field of elastic wave propagation, it is known that the presence of stiff inclusions in a matrix of a softer material may open a band gap due to scattering or resonant dynamics. The trees we found seem to be looking for patterns that fit this description: the deepest node encourages the existence of stiff inclusion while the first node encourages more soft material.

The unit-cell templates (Figure 3.4**b**) contrast with the shape frequency features in that they explicitly identify global (rather than local) patterns. In the unit-cell template sets for different frequency ranges, we can observe the existence of soft circles (closed curves) and stiff inclusions inside the circles, which are responsible for the formation of band gaps. As the frequency range moves higher, the size of the circle decreases. This supports the physical intuition that the smaller the stiff inclusions, the higher the frequency of the band gap.

3.6.4 Objective 4: Transfer to Finer Resolution

In Sections 3.5.1.3 and 3.5.2.4, we discussed how the proposed methods can transfer coarse scale information to finer resolution design space. To evaluate how well the model can transfer information, we trained the models on coarse resolution (10×10) unit-cells and tested them on finer resolution (20×20, 40×40 and 80×80) unit-cells. Table 3.1(c) shows the transfer precision of GOSDT+SFF and unit-cell template sets for band gaps in different frequency ranges. Other ML methods are not directly comparable because standard ML models trained on 10×10 data cannot take in 20×20 data. Therefore, we compared the proposed methods with baselines with slight modifications: we resized the fine resolution (20×20) unit-cell to the original size (10×10) and applied two algorithms (CNN or LightGBM) for rejection sampling. As before, if the CNN or LightGBM model predicts that the resized design has a band gap, we accept that sample, otherwise we reject it. The resizing was done via bicubic interpolation. Here, we did not compare with deep generative models such as GANs. Although GANs (which are notoriously hard to train) might generate materials faster than rejection sampling, their precision can only be lower because GANs' discriminators are co-trained with the generator, and thus cannot be more accurate than a CNN directly trained to predict only the target. For each frequency range, we asked the trained models to sample 200 unit-cell designs in finer resolution space, and ran the FEA simulation to obtain the true band gap property for evaluating the transfer precision. For the baseline models and GOSDT+SFF, we show the results for 20×20 design space. As unit-cell template sets performs extremely well on this task, and generates new designs efficiently, we also show its results in 40×40 and 80×80 design space. Please see Supplementary Information C for visual illustrations of how to sample fine resolution designs using each model.

The results in Table 3.1(c) indicate that **the unit-cell templates, when transferred to all finer resolutions** (20×20**,** 40×40 **or** 80×80)**, have very high precision, with almost no precision drop compared to** 10×10. GOSDT+SFF and other baselines do not generalize as well as unit-cell templates to the finer resolution design space. GOSDT+SFF performs better than the resizing baselines in [0, 10] kHz and [10, 20] kHz, but performs worse than baselines in [20, 30] kHz. Interestingly, the transfer precisions, of both GOSDT+SFF and unit-cell template sets, decrease as frequency ranges moves higher, although unit-cell template sets have a much slower precision decrease than GOSDT+SFF. A possible explanation for the decrease of precision is that, in higher frequency ranges, the band gaps are physically more related to finer scale features that are not included in the coarse resolution dataset. Since the models are trained on coarse resolution data, they can only transfer physics that occurs in coarse patterns to finer resolution design space, but cannot discover finer scale physics without supervision. But as shown by the transfer precision results, we should emphasize that our unit-cell template sets method was capable of extracting critical coarse resolution features such that this decrease of precision at finer resolution due to wave physics is minimized (the worst transfer precision is still above 90%). One further potential improvement to this is to add new samples at each finer resolution design space when transferring between extreme scales.

In Supplementary Information D.3, we show additional results on sampling with correlation between green pixels for unit-cell template sets, which demonstrates the surprising flexibility of unit-cell template sets in terms of designing at finer-resolution design space.

3.6.5 Practicality Test

In our method and simulations so far, we assumed the unit-cell is tiled infinitely for computational convenience. However, a unit-cell can only be tiled finitely in practice and the results can differ for infinite and finitely tiled domains due to boundary conditions. To test whether the designs found by our method work in practice, we simulated the dispersion relations of finitely-tiled materials made by unit-cell designs discovered by our method. See Supplementary Information D.4 for results of the finite tiling COMSOL simulation. The results show that our method is robust under finite tiling.

In addition to the finite tiling test, we also tested the practicality of the proposed method on its ability to create a wave demultiplexer (Figure 3.5), in which waves with different frequencies travel through the materials in different directions. That is, a signal enters the demultiplexer, and there are three different possible outputs; which one will be non-zero depends on the frequency of the input signal. Specifically, we will build the demultiplexer to route signals from three different frequency ranges in different directions.

We need 5 different materials to build the demultiplexer: one material with band gaps covering all three ranges, one material allowing band pass in all ranges, and three materials allowing band pass in one range while blocking the other two ranges. We use homogeneous stiff unit-cells for the material allowing band pass in all ranges. For other materials, we train a unit-cell template set to find 20×20 unit-cell designs with these properties, and assemble the 5 unit-cells to build the demultiplexer (top row of Figure 3.5). The bottom row of Figure 3.5 shows the how the demultiplexer successfully guides waves with different frequencies (11.45, 13.97 and 15.55 [kHz]) towards different directions, which was the goal of the experiment.

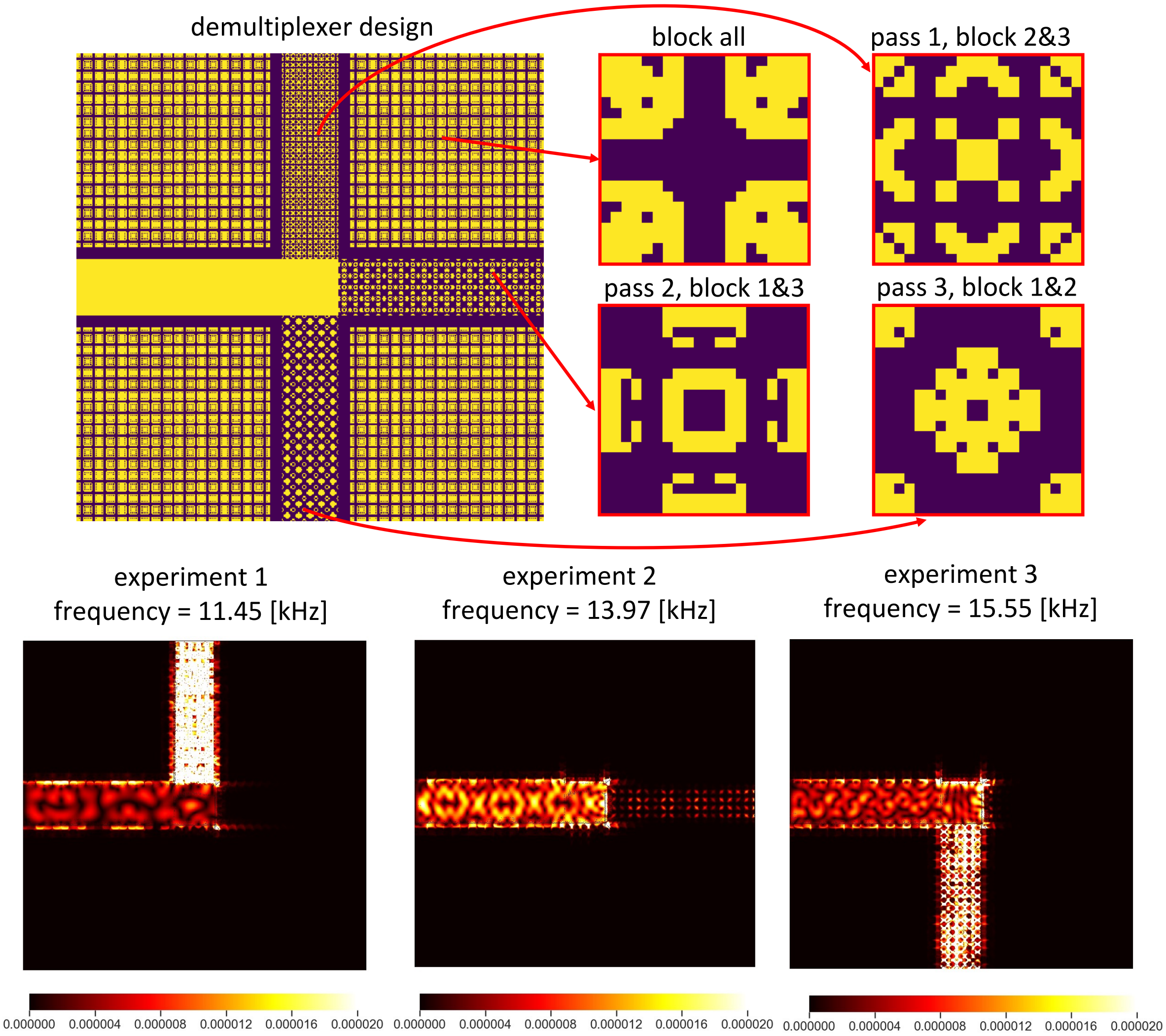


Figure 3.5: Mechanical wave demultiplexer and the displacement fields for input signals frequencies. Each part of the demultiplexer was built using our discovered unit-cells that have desired properties. *Top*: The demultiplexer and the unit-cell designs used to make the demultiplexer. *Bottom*: Magnitude of displacement fields when signals of different frequencies (11.45, 13.97 and 15.55 [kHz]) are fed into the left side of the demultiplexer. The displacement values are clipped if they exceed the display range. The signals with different frequencies go through different channels in the demultiplexer: 11.45 [kHz] goes upward, 13.97 [kHz] goes right, and 15.55 [kHz] goes downward, as desired. Note that, although the passing signal of experiment 2 is not as strong as signals in the other experiments, it still pass the channel on the right without fading inside the channel.

3.7 Conclusion and Discussion

Our work shows the power of interpretable machine learning tools in material design. The approach has achieved both mechanistic understanding, e.g., physically interpretable rules of patterns that lead to band gaps, and designing new materials with desired functionality. The approach has been demonstrated to be predictive for both infinite domains and realistic finite domains, and it has been able to design the material geometry for a wave demultiplexer. Additionally, our multi-resolution framework, which robustly carries coarse-scale knowledge to finer resolutions, is potentially applicable to a wide range of materials science problems.

Since our method learns robust coarse-scale features that can generalize to finer-resolution design space, it might also be useful in future studies for determining how to collect finer-resolution training data for rapidly capturing fine-scale physics. Using these new data, we might be able to fine tune the model so that it can more efficiently capture physics at multiple scales.

Data Availability

All electronic structure data and the simulation code developed in this paper are available upon request.

Code Availability

The code for replicating our results will be available upon acceptance.

Acknowledgements

The authors are grateful to M. Bastawrous, A. Lin, K. Liu, C. Zhong, O. Bilal, W. Chen, C. Tomasi, and S. Mukherjee for the feedback and assistance they provided during the development and preparation of this research. The authors acknowledge funding from the National Science Foundation under grant OAC-1835782, Department of Energy under grant DE-SC0021358 and DE-SC0023194, and National Research Traineeship Program under NSF grant DGE-2022040 and CCF-1934964.

Competing Interests

The authors declare no competing interests.

Chapter 4

GAUSSIAN PROCESS REGRESSION AS A SURROGATE MODEL FOR THE COMPUTATION OF DISPERSION RELATIONS

[1] Alexander C. Ogren et al. “Gaussian Process Regression as a Surrogate Model for the Computation of Dispersion Relations”. In: *Computer Methods in Applied Mechanics and Engineering* 420 (Feb. 15, 2024), p. 116661. ISSN: 0045-7825. DOI: `10.1016/j.cma.2023.116661`. URL: `https://doi.org/10.1016/j.cma.2023.116661` (visited on 04/14/2024).

Author contributions

A.C.O. conceptualized the project, wrote and revised of the manuscript (with greatly appreciated help from B.T.F, and suggestions from C.D. and K.L.B.), wrote the simulation code, developed and tested the surrogate model. B.T.F. performed the comparison against Convolutional Neural Networks.

Chapter preface

This chapter develops a surrogate model, *GPR-dispersion*, for the efficient computation of full dispersion relations. Dispersion relations, the key mathematical descriptors for wave propagation through media, are usually computed only partially, and are rarely computed in full due to their computational cost. This limitation confines the elastodynamic design of engineered materials to simple objectives, for example, one or two bandgaps or wave speeds. This work enables the efficient computation of full dispersion relations through a data-driven surrogate model based on Gaussian process regression, opening doors for the design of wave propagation in engineered materials in a truly multi-functional context.

Fundamentally, GPR-dispersion accelerates the computation of dispersion relations by reducing the number of wavevectors for which modal eigenvalue problems need to be solved. Importantly, this number is not reduced to zero, and GPR-dispersion still directly samples the physics of the problem through a purely physics-based model. While most Gaussian process models use standard *isotropic* covariance functions, GPR-dispersion innovates by employing an elegant and simple data-driven covariance function learned through direct covariance measurements. Of the desired attributes for surrogate models, as put forth in chapter 2, GPR-dispersion exhibits *transparent and principled modeling* through the direct use of physics-based dispersion computations, *representation invariance* through the thoughtful selection of representation-agnostic predictors, *inverse-design utility* through the provision of design gradients and uncertainty metrics, *ease of set-up and use* through data efficiency and an extremely inexpensive and deterministic training process, and demonstrates *generalizability* by performing well on out-of-distribution data.

Although GPR-dispersion was initially designed with the application of engineered materials in mind, it also may have utility in inference problems to accelerate fitting routines, as discussed in chapter 6.

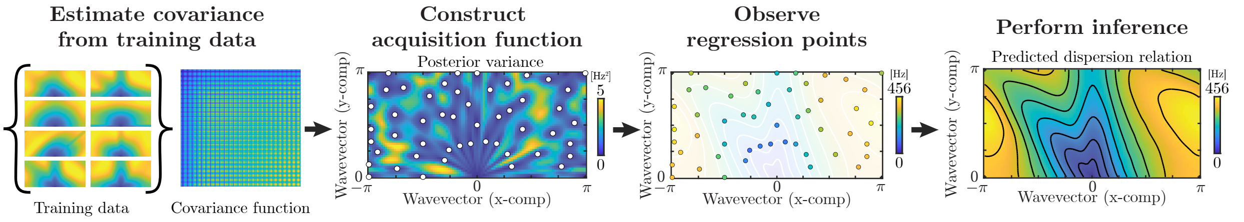


Figure 4.1: A graphical depiction of our surrogate model pipeline. The model first observes a dataset to construct the covariance function of the Gaussian process model. Using this covariance function, the model builds its acquisition function, determining the locations of the regression points it will sample. Finally, the model observes the value of the dispersion relation at the selected regression points, and infers the dispersion solution everywhere in the irreducible Brillouin zone. The surrogate model makes this inference two orders of magnitude faster than the exact calculation, and one order of magnitude faster than an equivalent-error linear interpolation model.

4.1 Abstract

The ability to design materials for wave propagation behaviors has high potential for impact in medical imaging, telecommunications, and signal processing. The dispersion relation is the key mathematical object that describes linear elastic wave propagation behavior in a material. To date, the design of wave propagation behavior in materials has been limited to one or two design objectives, for example, designing for one or two bandgaps or wavespeeds. This is a result of the complicated relationship between a material and its dispersion relation, and the hefty computational cost of traditional dispersion computations. Decreasing the amount of time required to perform dispersion computations will streamline the design process of wave devices, allowing for more thorough design optimization workflows, more highly resolved design features, and multi-functional materials.

In this work, we demonstrate a specialized Gaussian Process Regressor as a surrogate model for the computation of dispersion relations to alleviate the immense computational cost of the traditional model. By measuring covariance information from relatively small training sets (10s to 1000s of dispersion relations), our surrogate model efficiently infers the full dispersion relation after solving for the dispersion relation at only a sparse set of wavevector points. Further, we provide a

mathematical framework to use this model for material design through gradient-based optimization methods.

4.2 Introduction

The structure and constitution of a material dictate the properties of the waves it can support, which are thoroughly described by the material's dispersion relation. A material's dispersion relation is the relationship between the frequencies and wavevectors of all linear elastic waves that can travel in the material. Dispersion relations contain information about bandgaps (frequencies at which waves experience spatial attenuation), group velocity (the velocity with which energy can be transported by waves in the material), and phase velocity (the velocity at which a wave peak will move in the material). Bandgap information can be leveraged to engineer filtering and reflection, and group velocity and phase velocity information can be leveraged to achieve refraction. These fundamental behaviors can in turn be utilized to design complex wave propagation devices such as wave guides[14, 126, 127], wave lenses [15, 128, 129], wave multiplexers [14, 126, 130], and more.

Current methods to calculate dispersion relations are based on numerically computing the eigenvalues of the elastic wave equation [131]. Computational limitations are especially restrictive when these expensive forward models are used to solve the inverse problem of design [16, 132–135]. As a result, scientists and engineers are often forced to simplify design workflows by limiting the design space to be relatively narrow, considering only partial dispersion information and/or relatively basic design objectives [14–17, 126, 132, 135].

Our method aims to increase the accessibility of full dispersion relations to alleviate the need for simplifications such as partial dispersion computations. Full dispersion relations are defined over a region of reciprocal space (wavevector space) called the *irreducible Brillouin zone* (IBZ). Due to computational costs, dispersion relations are often computed only along one-dimensional slices of these regions called *irreducible Brillouin zone contours* (IBZ contours). However it's been recently shown that using this simplification can lead to errors [136]. Further, plane symmetry groups without mirror symmetries do not have a uniquely defined IBZ contour. For unit cells belonging to these plane symmetry groups, dispersion analysis along a contour will yield different results depending arbitrarily on the choice of contour. As a second example, a common shortcut when designing for group velocity is to compute the dispersion relation at only a handful of wavevectors or frequencies of interest. This is useful for achieving desired wave behavior in specific directions of propagation [16], or at specific frequencies of interest [17]. Although these methods are sufficient for designing simple wave devices, the design of next-generation, multi-functional wave devices will require the efficient computation of full dispersion relations.

Techniques have been developed specifically to efficiently solve dispersion problems such as Bloch mode synthesis [137, 138], spectral X-FEM [139], specialized mixed-variational formulations [140], reduced Bloch mode expansion [141], reduced Bloch operator finite element methods [142], and wave finite element methods (WFEM) [143, 144]. However, even these specially tailored methods executed by a highly parallelized and optimized implementation on a multi-CPU multi-GPU cluster admit the need for more efficient ways to explore the design space [135]. To perform design optimization on a 48^3 voxel domain, a distributed computing setup using 16 NVIDIA GTX-780 GPUs takes 6.5 minutes per optimization iteration, even when computing the dispersion relation only along the IBZ contour rather than over the full IBZ [135]. Fundamentally, each of [137–144] reduces the cost of dispersion computations by reducing the number of effective degrees of freedom involved in the model, thereby reducing the size of the eigenvalue problem that needs to be solved at each wavevector point. In contrast, our surrogate model reduces the total number of eigenvalue problems that need to be solved. Due to this construction, our surrogate model can seamlessly integrate with any of these aforementioned techniques to produce an efficiency improvement that compounds with that of the physics-based technique being paired with our model.

Surrogate models, often involving machine learning, have been proposed in a variety of settings to reduce the computational cost of purely physics-based methods. The applications range from multi-scale modeling [145–148], to dynamics [149], statics [150], turbulence [151], optics [152], and many others. For dispersion relation computations specifically, a fixed-resolution convolutional neural network (CNN) surrogate model has been proposed [84]. This CNN model is shown to vastly outperform the traditional densely connected neural network, suggesting that the convolutional architecture is more appropriate for these types of problems. Because CNNs can make predictions so quickly once the model is trained, it seems like a very attractive option. However, neural networks often require large amounts of training data, which are expensive to compute (hundreds of thousands of precomputed dispersion relations [84]) and require a lengthy training period (13 minutes on a GPU reported just for the 1D problem [84]). Unless the trained model is subsequently used to make predictions for more dispersion relations than were included in the training set, the time used to generate the training data outweighs the time saved by using the model. Further, the CNNs architecture locks the user into a fixed unit cell resolution and fixed dispersion relation resolution. If a user desires to change the resolution of the discretization of the unit cell or IBZ, a new model architecture needs to be trained on a new training set with the corresponding resolution. Recently, there has been a push towards surrogate models that exhibit resolution independence, meaning that the model can train on data of one spatio-temporal resolution, and make predictions at another resolution [153–155]. In the spirit of this effort, the surrogate model we propose exhibits resolution independence both in terms of unit cell discretization and IBZ discretization.

We propose using Gaussian process regression [156] to obtain principled surrogate models of full dispersion relations. In particular, we propose a data-driven covariance function that only requires a small training set of dispersion relations (tens to thousands) and allows for an efficient acquisition function. In comparison to deep learning-based surrogate models, our model achieves higher accuracy with less training data and is invariant to unit cell representation.

The key features of our model are:

1. The model is independent of unit cell representation. For example, if the model learned covariance information from a dataset of parametrically defined unit cells, it can still be used to make predictions for pixelated designs, and vice versa.
2. The model requires a relatively low amount (tens to thousands) of training data samples to make accurate predictions. The model learns from data by measuring covariance information, which takes less than 20 seconds even for reasonably large training set sizes.
3. The model uses a physics-based method to sample the dispersion relation at various points in the IBZ. This physics-based method is a modular component of the model that can be swapped for other, faster physics-based methods for dispersion computations. Because of this, the efficiency gains provided by our model directly augment the efficiency gains of the physics-based model.
4. The model not only supplies predictions, but also offers uncertainty estimates.
5. The model is differentiable, allowing for its use in gradient-based topology optimization.

In this paper, we first discuss the traditional methods currently used for the computation of dispersion relations and gradient-based optimization of dispersion relations. We then go on to provide a brief overview of Gaussian process regression (GPR). Next, we describe the details of our model and demonstrate how it would be applied to one-dimensional problems. Finally, we apply our model to two-dimensional unit cells and evaluate its performance against traditional methods as well as another existing surrogate model.

4.3 Traditional computation of dispersion relations

To compute the dispersion relation of a periodic structured material, the physical domain of analysis is reduced to a single unit cell. The frequency domain dynamics of linear-elastic periodic solids are governed by the harmonic elastic wave equation with Bloch-Floquet periodic boundary conditions. The harmonic elastic wave equation looks for wave mode solutions that obey linear elasticity, and assumes that these solutions will be temporally sinusoidal. The Bloch-Floquet boundary conditions enforce that the solution is periodic (since the domain is periodic), but with a complex phase offset. The complex phase offset is used to make sure we are finding modal wave solutions with a variety of different wavelengths (not just wavelengths that are integer divisors of the unit cell's dimensions).

The harmonic elastic wave equation in D dimensions is

$$-\rho(x)\omega^2 u_i = \frac{\partial}{\partial x_j}\left[C_{ijkl}(x)\frac{1}{2}\left(\frac{\partial u_k}{\partial x_l}+\frac{\partial u_l}{\partial x_k}\right)\right], \quad i,j,k,l = 1,...,D, \tag{4.1}$$

and the Bloch-Floquet periodic boundary conditions are

$$u(x+a_n) = u(x)e^{\mathbf{i}\gamma\cdot a_n}\ \forall x \in \{x \in \Theta \mid x + a_n \in \Theta\}, \quad n = 1,...,D. \tag{4.2}$$

The density, $\rho(x)$, and the material stiffness tensor, $C_{ijkl}(x)$, define the linear-elastic material properties of the unit cell, Θ. The wavevector, γ, and the eigenfrequency, ω, inversely describe the length and time scales of the wave. The displacement field, $u(x)$, describes the vibrational mode by which the wave travels. Finally, the lattice vectors, a_n, define the lattice of the periodic material, and $\mathbf{i}$ is the imaginary unit. The equation is a generalized eigenvalue problem with $u(x)$ and ω^2 as the eigenvector and eigenvalue respectively. To compute the dispersion relation, the eigenvalues of this equation must be determined as a function of γ, $\forall\gamma \in \Gamma$ (letting Γ denote the irreducible Brillouin zone).

With the exception of a few simple cases, analytical solutions for dispersion relations are not obtainable. For this reason, numerical solutions are often used to discretize the generalized eigenvalue problem (Eq. 4.1). Although there are many numerical methods to compute dispersion relations, one of the most well-known (though not necessarily the most efficient) is the Finite Element Method. Within the framework of FEM, the dispersion relation can be computed by assembling the stiffness and mass matrices, $K(\gamma)$, and $M(\gamma)$, and solving the generalized eigenvalue problem for each γ in a discretization of the IBZ:

$$\left[K(\gamma) - \omega(\gamma)^2 M(\gamma)\right] u(\gamma) = 0. \tag{4.3}$$

There are several other specially tailored variational methods that have been shown to converge faster than the finite element method when solving this particular PDE [140]. We note that the surrogate model we have developed can be used in conjunction with any of these more highly-specialized variational methods.

4.4 Traditional gradient-based optimization of dispersion relations

Gradient-based optimization is a powerful design tool that can be used in many contexts (including wave propagation [16, 132, 135]) to obtain material designs that exhibit desired behaviors. Designs are represented by a design vector, θ, containing parameter values that uniquely determine the geometry and constitutive properties of the unit cell design. The goal of design optimization is to determine a θ that results in the desired wave behavior, often achieved by maximizing or minimizing an objective function (which encodes design objectives) via gradient-based methods. To use these gradient-based methods, gradients of the objective function are needed, which in turn depend on gradients of quantities embedded in the dispersion relation. In this section, we quickly step through a few known equations commonly used in optimization problems surrounding wave behavior. In this context, K and M are not only functions of the wavevector, but also of the design vector, θ. Derivations of these equations can be found in 4.10.3.

To modify the frequency, ω, at a specific wavevector by incrementally changing the design variable, the following sensitivity equation is useful. Assume all eigenvectors are normalized by the mass matrix; i.e., $u^*Mu = 1$ where $(\cdot)^*$ denotes conjugate transpose. For ease of reading, assume here that θ is a scalar. These equations can easily be extended to vector θ by repeating the following computations for each entry of θ. The gradient of one frequency in the dispersion relation with respect the design variable is given by

$$\frac{\partial \omega}{\partial \theta} = \frac{1}{2\omega} u^* \left[\frac{\partial K}{\partial \theta} - \omega^2 \frac{\partial M}{\partial \theta} \right] u. \tag{4.4}$$

To modify the group velocity, $\frac{\partial \omega}{\partial \gamma}$, at a specific wavevector, the sensitivity computation is more complicated and more computationally expensive. The group velocity can be computed analytically by

$$\frac{\partial \omega}{\partial \gamma} = \frac{1}{2\omega} u^* \left[\frac{\partial K}{\partial \gamma} - \omega^2 \frac{\partial M}{\partial \gamma} \right] u. \tag{4.5}$$

The design sensitivity of the group velocity can be computed analytically by

$$\frac{\partial}{\partial \theta} \left[\frac{\partial \omega}{\partial \gamma} \right] = \frac{\partial^2 \omega}{\partial \theta \partial \gamma} = -\frac{1}{2\omega^2} \frac{\partial \omega}{\partial \theta} u^* A u + \frac{1}{2\omega} \frac{\partial u^*}{\partial \theta} A u + \frac{1}{2\omega} u^* \frac{\partial A}{\partial \theta} u + \frac{1}{2\omega} u^* A \frac{\partial u}{\partial \theta} \tag{4.6}$$

where $A = \frac{\partial K}{\partial \gamma} - \omega^2 \frac{\partial M}{\partial \gamma}$, and $\frac{\partial u}{\partial \theta}$ can be computed by solving the linear system:

$$\begin{bmatrix} \mathrm{Re}[K - \omega^2 M] & -\mathrm{Im}[K - \omega^2 M] & \mathrm{Re}\,[-Mu] & -\mathrm{Im}[-Mu] \\ \mathrm{Im}[K - \omega^2 M] & \mathrm{Re}[K - \omega^2 M] & \mathrm{Im}\,[-Mu] & \mathrm{Re}\,[-Mu] \\ \mathrm{Re}\,[u^*M] & -\,\mathrm{Im}\,[u^*M] & 0 & 0 \\ 0 \cdots 0 & 1\, 0 \cdots 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \mathrm{Re}[\frac{\partial u}{\partial \theta}] \\ \mathrm{Im}[\frac{\partial u}{\partial \theta}] \\ \mathrm{Re}[\frac{\partial \mu}{\partial \theta}] \\ \mathrm{Im}[\frac{\partial \mu}{\partial \theta}] \end{Bmatrix} = \begin{Bmatrix} \mathrm{Re}[-\left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u] \\ \mathrm{Im}[-\left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u] \\ \mathrm{Re}[\frac{1}{2} u^* \frac{\partial M}{\partial \theta} u] \\ 0 \end{Bmatrix} \tag{4.7}$$

Not only is it computationally expensive to compute dispersion relations, but it is also computationally expensive to compute the design sensitivities of dispersion relations. Lu et al. report sensitivity analysis accounting for nearly half the time consumed by one iteration of optimization [135]. Later, we will explain how our surrogate model gives multiple options for computing sensitivities, opening the possibility for further computational savings. We also provide explicit equations for how to compute these gradients when using the surrogate model.

4.5 Gaussian process regression

4.5.1 Overview

Gaussian process regression (GPR) is a probabilistic regression method that constructs a predictive distribution for an unknown function, $f(x)$, by combining observations of the function with prior information [156]. Because the function is unknown, we think of $f(x)$ as being a random function distributed according to a Gaussian process $\mathcal{GP}(m(x), k(x, x'))$ defined by mean function $m(x)$ and covariance function $k(x, x')$.

$$f(x) \sim \mathcal{GP}(m(x), k(x, x')) \tag{4.8}$$

Evaluation of this function at any finite number of points is described by a joint Gaussian distribution. Let m and k operate row-wise when given matrix input, with $m(X)$ yielding the mean vector corresponding to the points indicated by the rows of X, and $k(X, X')$ yielding the covariance matrix for the points in X and X'. With matrix X_* and vector f_* denoting the set of points for which we would like to make evaluations, the prior distribution is

$$f_* \sim \mathcal{N}\left(m(X_*), k\left(X_*, X_*\right)\right) \tag{4.9}$$

where $\mathcal{N}$ denotes a normal distribution.

Given a dataset $R = \{(x_i, y_i) | y_i = f(x_i) + \varepsilon, \varepsilon \sim \mathcal{N}(0, \sigma_f^2)\}_{i=1}^N$ of (possibly noisy) observations of $f(x)$, the predictive distribution is formed by conditioning the prior distribution on the observations. With X and y denoting the observations, and I denoting the identity matrix, the predictive distribution is

$$\begin{aligned} f_* \mid X_*, X, y \sim \mathcal{N}\big(m(X_*) + k\left(X_*, X\right) & \left[k(X, X) + \sigma_f^2 I\right]^{-1} y, \\ k\left(X_*, X_*\right) - k\left(X_*, X\right) & \left[k(X, X) + \sigma_f^2 I\right]^{-1} k\left(X, X_*\right)\big). \end{aligned} \tag{4.10}$$

Numerically, Eq. 4.10 is often implemented in two steps. The first step is to find the model weights, α, which are independent of the prediction locations, by solving the linear system in Eq. 4.11.

Subsequently, predictions can be made at arbitrary locations X_* using the pre-computed model weights α in Eq. 4.10 rather than solving the system anew.

$$\alpha = [k(X,X) + \sigma_f^2 I]^{-1} y \tag{4.11}$$

Because GPR yields a predictive *distribution*, we not only infer a prediction as the mean of the distribution, but also a measure of uncertainty as the variance of the distribution.

4.5.2 Eigenfunction analysis of the covariance function

In later sections, we perform eigenfunction analysis of our proposed covariance function. Eigenfunction analysis is often used to assess the convergence and expressive power of a Gaussian process regressor. A function $\phi(x)$ is an eigenfunction of the covariance function $k(x,x')$ with eigenvalue λ, with respect to measure $\mu(x)$ if it satisfies

$$\int k\left(x,x'\right)\phi(x)d\mu(x) = \lambda\phi\left(x'\right). \tag{4.12}$$

These eigenfunctions form a basis which spans the space of all possible functions that can be attained by the mean function of the Gaussian process regressor, which is called the *hypothesis space*. A Gaussian process regressor can exactly represent any function in the span of the eigenfunctions with nonzero eigenvalues. A covariance function with an infinite number of nonzero eigenvalues is said to be *nondegenerate*, while one with only a finite number of nonzero eigenvalues is said to be *degenerate*.

A Gaussian process regressor with a nondegenerate covariance function has almost unlimited expressive power since its hypothesis space is the span of an infinite eigenbasis. In the limit of infinite observations, the model can exactly fit any function that can be represented by the generalized fourier series $\sum_{i=1}^{\infty} a_i\phi_i(x)$. In other words, for a GP regressor with a nondegenerate covariance function, the posterior mean function will converge to the expected value of the true underlying distribution as the number of observations goes to infinity (assuming the true underlying distribution is reasonably well-behaved).

In contrast, a Gaussian process regressor with a degenerate covariance function has limited expressive power, since its eigenfunctions span only a finite dimensional hypothesis space. For a GP regressor with a degenerate covariance function, in the limit of infinite observations, the posterior mean function will instead converge to the best μ-weighted L^2 approximation of the expected value of the true underlying distribution within the span of the eigenfunctions. Practically, this means that although a Gaussian process regressor with a degenerate covariance function may not converge to the exact solution in the limit of infinite regression points, it can still make a highly accurate prediction if its eigenbasis is comprehensive enough for the desired application.

4.6 Methods

In this section, we give the specifics of our surrogate model, and demonstrate its application to relatively simple one-dimensional dispersion computations. This toy problem is meant only to serve as an example - the substance of our results can be found in the following section. Traditional dispersion computations require the solution of an eigenvalue problem at each wavevector in a discretization of the irreducible Brillouin zone (IBZ). The goal of our surrogate model is to predict the eigenvalue solutions for each wavevector after solving the eigenvalue problems at only a small subset of these wavevectors. For most unit cells, which have many physical degrees of freedom, it's common in dispersion analysis to only consider the smallest N eigenvalues. In our surrogate model framework, we use N separate GPR models to fit the N eigenvalue bands, $\{\omega_i(\gamma)\}_{i=1}^N$.

The structures used in this section for demonstration are periodic one-dimensional bars. They can only transmit waves along their axis and have relatively simple dispersion relations. We randomly generate unit cell designs which have smoothly varying constitutive properties, except at randomly selected interfaces where the property values are allowed to be discontinuous (details in 4.10.1). These interfaces lead to wave scattering behavior in the material – a mechanism well-known to cause band gaps in the dispersion relation. Examples of these unit cells and their dispersion relations are shown below in Fig. 4.2.

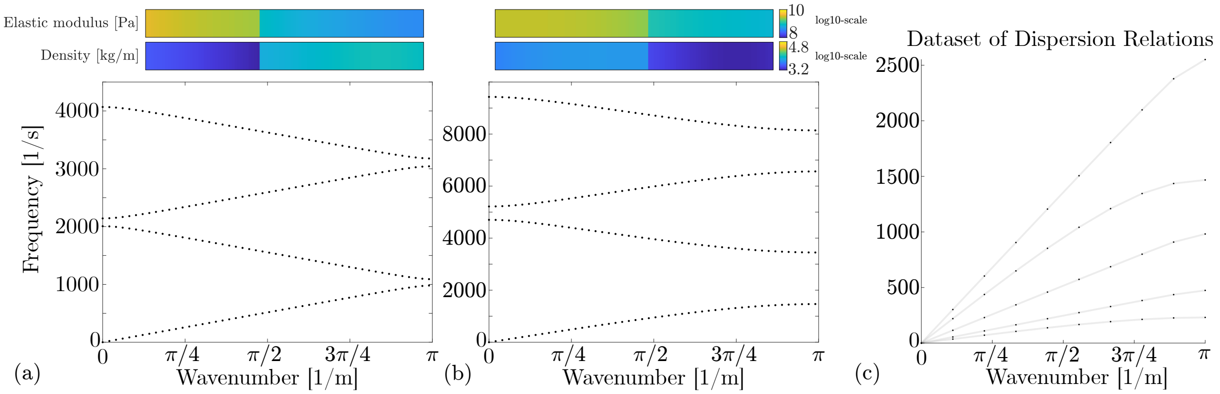


Figure 4.2: (a) and (b) Two different 1-D unit cell designs and the first four bands of each design's dispersion relation, computed on a 50-point discretization of the IBZ. Each design in this figure has two interfaces of material property discontinuity (one in the interior of the unit cell and one at the boundary). (c) A dataset of the first band of the dispersion relations computed for 5 random 1-D structures drawn from the same distribution as the designs in (a) and (b), evaluated on a 10-point discretization of the IBZ.

4.6.1 Mean function

Our model uses a zero mean function ($m(x) = 0$). This is common in GPR, and does not mean that the predictive distribution will have a zero mean. Nonzero mean functions are often used when the function to be fit has an underlying trend that can be written explicitly (such as a signal that, on average, increases linearly with time but has unpredictable variations around that average). Because dispersion relations have no such trends that can be explicitly written, we leave our mean function as zero. This leaves the entirety of the model's capacity to be carried by the covariance function.

4.6.2 Covariance function

The covariance function is usually the most crucial component in any Gaussian process model because it encodes the statistical coupling between function values at each point in the domain, defining underlying patterns and relationships in the data. Often, Gaussian process models use simple *isotropic* covariance functions such as the squared exponential or the Matérn covariance functions. Isotropic covariance functions are functions only of the distance between points in input space, meaning that they can only encode assumptions about the underlying function globally. These functions offer a convenient starting point for basic applications of Gaussian process modeling; however, they lack the expressiveness that can be offered by more thoughtful constructions of the covariance function. We propose a non-isotropic covariance function that encodes richer covariance information in a principled way that varies throughout the domain.

Our surrogate model uses a dataset of precomputed dispersion relations to construct its covariance function. Given a dataset $T = \{(\gamma, \omega)_i\}$ of dispersion relations evaluated at a fixed set of wavevectors Γ_T, we measure the covariance between the frequency values at all pairs of wavevectors in Γ_T. This covariance computation is extremely fast, especially when compared to the training process of neural networks. For example, to compute the covariance matrix for 10,000 dispersion relations with 10,000 points evaluated in each dispersion relation, only 6 seconds on a 9th generation Intel Core i7-9750H are required. Once the covariance matrix is computed, we use cubic interpolation of the measured covariance values to define the covariance function, $k : \Gamma^p \times \Gamma^q \to \mathbb{R}^{p \times q}$ mapping any two lists of wavevectors to the appropriate covariance matrix. Each eigenvalue band is described by its own separate covariance function. Although this data-driven covariance function is degenerate (i.e., in the limit of infinite regression points, the estimated dispersion relation converges to the best solution spanned by the finite eigenbasis rather than the exact numerical solution), it achieves higher accuracy than a data-agnostic covariance function given a limited number of regression points. Convergence to the exact numerical solution is not necessary in practical design problems, since even the exact solution does not perfectly represent real-world physics.

To demonstrate how this works for the one-dimensional case, we compute the covariance between frequency values at each pair of wavenumbers for the dataset shown in the right-most panel of Fig. 4.2. The results of this computation are shown in Fig. 4.3. For this demonstration, the dataset is small, with only 5 dispersion relations, and coarse, with the domain discretized into only 10 points.

Even so, we can see that we've already captured critical information about the type of function we are trying to fit. For example, we see that the variance is 0 at $\gamma = 0$, and that the frequency at this wavenumber has no covariance with the frequencies at any other wavenumber. This makes sense because, for the first eigenvalue band, this frequency is always zero, representing a rigid body mode, regardless of the rest of the dispersion relation. Conversely, the variance is maximum at $\gamma = \pi$, and the frequency at this wavenumber is covariant with the frequency at many other wavenumbers, especially those nearby.

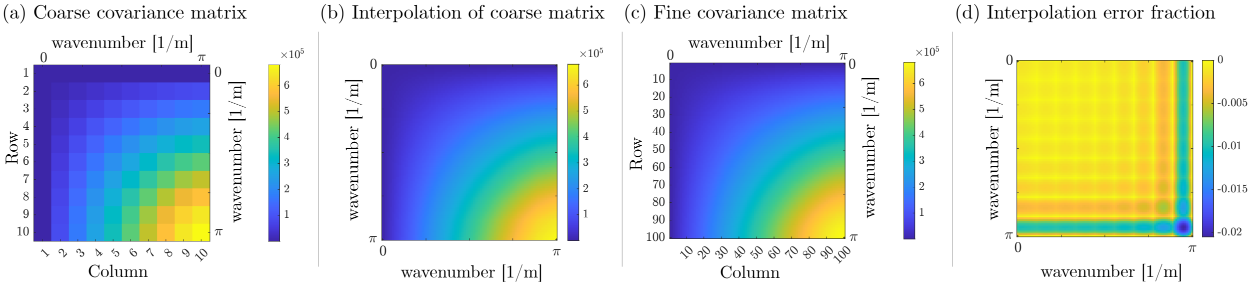

(a) Coarse covariance matrix (b) Interpolation of coarse matrix (c) Fine covariance matrix (d) Interpolation error fraction

Figure 4.3: (a) The covariance matrix computed directly from the coarse version of the dataset, using a 10-point discretization of Γ. (b) The covariance function generated by applying interpolation to the coarse covariance matrix. (c) the covariance matrix computed from the high resolution version of the same dataset, but using a 100-point discretization of Γ. This panel represents a visualization of the true covariance function. (d) The error fraction induced by approximating the high resolution covariance matrix with interpolation of coarse resolution covariance matrix. The error fraction is small, achieving a maximum absolute deviation of about 2%.

In later sections, when working with 2D structures, there are two wavevector components to consider. In these cases, the covariance function should be thought of as a continuous scalar-valued function of four scalar variables, and covariance interpolation must happen in this four-dimensional space. Regardless, for all computational operations other than interpolation, standard matrix operations can still be applied after reshaping the four-dimensional covariance array into a matrix.

With the covariance and mean functions defined, our Gaussian process prior is defined. We can visualize our prior by sampling it. With the exception that the samples are allowed to assume negative values, they have characteristics similar to the dispersion bands (see the first panel of Fig. 4.4). This shows that our covariance function yields a prior that encodes an appropriate hypothesis space, even before sampling any regression points.

4.6.3 Acquisition function

To fit the model to a specific dispersion relation, we supply it with a set of points (*regression points*) sampled from that dispersion relation. Because each regression point has a computational cost, it is beneficial to try to minimize the number of regression points needed, and maximize the information that each regression point contributes. An *acquisition function* is a function that tells the model how to select the locations of regression points. Here, we propose a greedy acquisition function that sequentially determines the most informative next point to sample. The acquisition function we have constructed uses the covariance function to evaluate the posterior variance over a discretization of the IBZ. The variance can be computed by evaluating the diagonal entries of the posterior covariance, $C_{posterior}$, given by

$$C_{posterior} = k\left(\Gamma_*, \Gamma_*\right) - k\left(\Gamma_*, \Gamma_R\right)\left[k(\Gamma_R, \Gamma_R) + \sigma_\omega^2 I\right]^{-1} k\left(\Gamma_R, \Gamma_*\right) \tag{4.13}$$

where Γ_* are the wavevectors constituting the discretization of the IBZ, Γ_R are the wavevectors representing the locations of the optimal regression points, and I denoting the identity matrix.

Starting with no regression points (Γ_R is empty, and the entire second term is zero), we consider the wavevector in Γ_* with the highest variance to be the most informative next wavevector to sample, so it becomes the next regression point location. We append this wavevector to Γ_R, re-evaluate $C_{posterior}$, and repeat. This process can be iterated until a desired number of regression point locations is obtained.

When this process is finished, Γ_R represents an ordered list of optimal regression points. During the fitting process (described in section 4.6.4), subsets of Γ_R can be selected by taking the first N elements if the computational budget does not permit the solution of the phononic eigenvalue problem for every wavevector in Γ_R. Because our covariance function is degenerate, there is a finite limit on the number of regression points that can be effectively used by the model, and Γ_R will not be able to grow beyond this size (the inversion in Eq. 4.13 will run into conditioning issues). This limit can be quite large, and depends on the size and resolution of the dataset that was used to construct the covariance function.

Conveniently, our acquisition function is entirely independent of the observed values of the regression points. This means the locations (wavevectors) of the optimal regression points can be determined ahead of time without actually making observations of the regression points themselves, allowing for the parallel computation of regression point values (frequencies) later. Additionally, this means that the final Γ_R is dependent only on the covariance function, and does not change as a function of the specific dispersion relation that the model is being fit to.

We visualize how our acquisition function works for a simple one-dimensional problem in the bottom row of panels of Fig. 4.4 below. Without performing eigenvalue computations, we iteratively evaluate the posterior variance, and append the point of maximum variance to the ordered list of regression points.

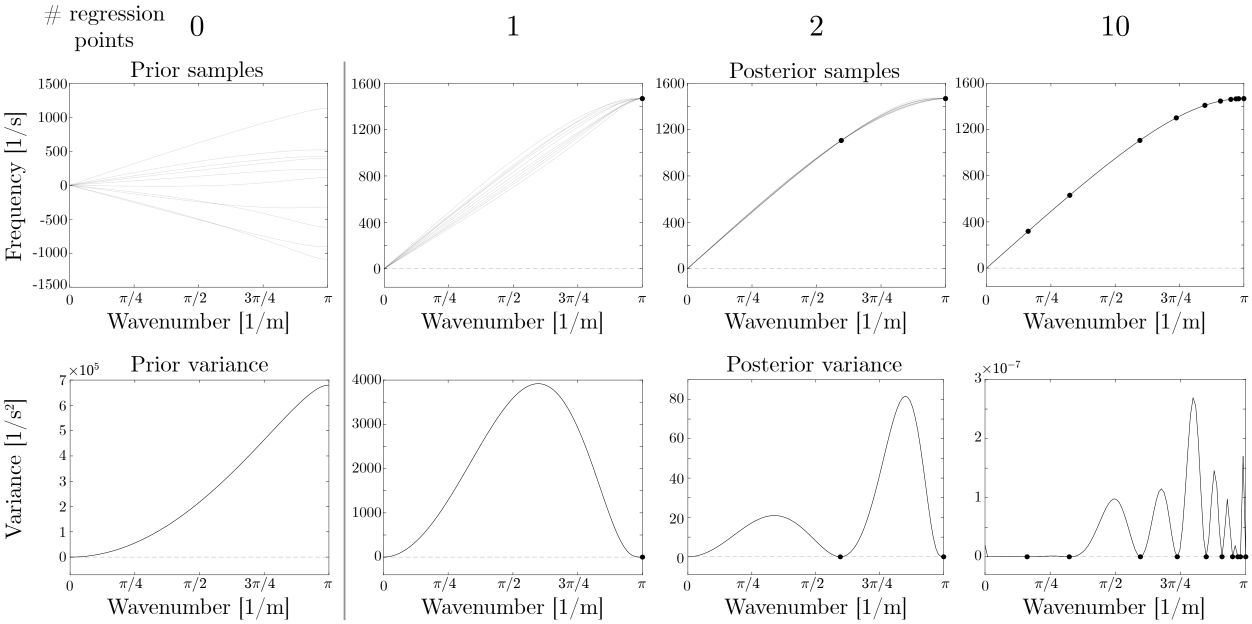


Figure 4.4: Top row: samples generated from the posterior (predictive) distribution with 0, 1, 2, and 10 regression points sampled (note that when 0 points are sampled, the posterior is the prior). As more points are sampled, the posterior distribution becomes much tighter, as can be seen by the decrease in variation of samples drawn from the posterior distribution. Bottom row: the variance of the posterior distribution with 0, 1 2, and 10 points sampled. The first three panels of the bottom row demonstrate the first three sequential steps of the algorithm used to determine the order in which points are sampled.

4.6.4 Fitting the model

With the mean, covariance, and acquisition functions determined, we can fit the GP model to a specific dispersion relation. To fit the model, we supply it with a set of regression points sampled from that dispersion relation. Our acquisition function determines the locations (wavevectors) of these points ahead of time. To perform the fit, we need to *observe* the eigenfrequency solutions associated with these wavevectors by solving the corresponding eigenvalue problem. Because the locations of the regression points have been determined ahead of time, these computations can be done in parallel. After observing regression points (Γ_R, Ω_R), we compute the model weights α by solving the linear system

$$\alpha = [k(\Gamma_R, \Gamma_R) + \sigma_\omega^2 I]^{-1}\Omega_R \tag{4.14}$$

where Γ_R and Ω_R are the wavevectors and frequencies of the regression points, σ_ω^2 is the assumed variance of the noise on the regression point measurements, and I is the identity matrix. Because our regression points are generated by a well-chosen finite element model, it is reasonable to assume values of σ_ω^2 as low as 0, if assuming the eigenvalue solutions are exact, or 10^{-16} to represent the uncertainty associated with using double precision floating point data format. We find that using

$\sigma_\omega^2 = 0$ works well for our purposes, and doesn't produce significantly different behavior from using $\sigma_\omega^2 = 10^{-16}$.

Once the model is fit to the regression points, the predictive distribution can be computed (Eq. 4.15). Because the predictive distribution is Gaussian, the maximum a posteriori estimate of the full dispersion relation is the mean of the predictive distribution. Further, the diagonal of the covariance matrix of the predictive distribution is the variance of each point being predicted. The variance gives us a metric for the model's predictive uncertainty at each wavevector in Γ_*. The predictive distribution is

$$\begin{aligned}\Omega_* \mid \Gamma_*, \Gamma_R, \Omega_R \sim \mathcal{N}\big(&k\left(\Gamma_*, \Gamma_R\right)\alpha, \\ &k\left(\Gamma_*, \Gamma_*\right) - k\left(\Gamma_*, \Gamma_R\right)\left[k(\Gamma_R, \Gamma_R) + \sigma_\omega^2 I\right]^{-1} k\left(\Gamma_R, \Gamma_*\right)\big)\end{aligned} \tag{4.15}$$

where Γ_* are the wavevectors at which we would would like to predict the frequency values of the dispersion band, and Ω_* are the predicted frequencies. In the top row of Fig. 4.4, we draw samples from this predictive distribution to show its evolution as more regression points are used. When fewer regression points are used, the predictive distribution is quite broad, indicating uncertainty in the prediction. However, as more regression points are used, the predictive distribution becomes extremely tight, indicating much more certainty.

4.6.5 Gradients for gradient-based optimization

So far, we have detailed how to make predictions with our model. However, to use our model in gradient-based design optimization, we must be able to supply efficient design sensitivity computations. In this subsection, we describe how to compute gradients for frequency optimization and group velocity optimization.

4.6.5.1 Gradients for frequency optimization

Typically, computing the design sensitivity of the frequency at one wavevector requires not only the eigenvalue solution, but also the eigenvector solution (Eq. 4.4). However, our model predicts only the eigenvalue, not the eigenvector. There are at least two ways to amend this.

First, we could simply use the surrogate model to infer the dispersion relation, and use this inference to determine the wavevector at which we need to optimize the frequency. Then we could solve the eigenvalue problem corresponding to that single wavevector, and compute the design sensitivity using Eq. 4.4. This is a great option if we only need to compute the design sensitivity of the frequency at a single wavevector, but do not know a priori at which wavevector (e.g. for bandgap optimization).

Alternatively, if the eigenvalue problems are particularly expensive, or if there are many wavevectors at which we desire to optimize the frequency, we can use a trick to avoid solving more eigenvalue problems. By employing the chain rule we can compute the gradient of any set of frequencies without performing additional eigenvalue computations. To do this, we first compute the gradient of the frequency at any wavevector with respect to the frequencies measured at the regression points. This gradient is defined by the GPR model. Then, since we do know the eigenvector at the regression points, we compute the gradient of the frequency at the regression point with respect to the design variables. This second gradient is defined by Eq. 4.4. We then multiply these two quantities to obtain the design sensitivity of any frequencies we are interested in optimizing. That is,

$$\frac{\partial \Omega_*}{\partial \theta} = \frac{\partial \Omega_*}{\partial \Omega_R} \frac{\partial \Omega_R}{\partial \theta} \tag{4.16}$$

where

$$\frac{\partial \Omega_*}{\partial \Omega_R} = k\left(\Gamma_*, \Gamma_R\right)\left(k(\Gamma_R, \Gamma_R) + \sigma_\omega^2 I\right)^{-1} \tag{4.17}$$

and

$$\left(\frac{\partial \Omega_R}{\partial \theta}\right)_{i,j} = \frac{1}{2(\Omega_R)_i}(U_R^*)_{i,:}\left[\frac{\partial K}{\partial \theta_j} - (\Omega_R)_i^2 \frac{\partial M}{\partial \theta_j}\right](U_R)_{:,i} \tag{4.18}$$

where U_R is a matrix containing the eigenvectors of the regression points, $(\cdot)^*$ denotes conjugate transpose, and the colon subscript indicates every element along the corresponding axis of the array (e.g. $(U_R)_{:,i}$ represents the i^{th} column of U_R). Note that although this method does not require further eigenvalue solutions, it does require traditional sensitivity computations for a potentially large number of regression points. In practice, these two costs should be weighed against each other.

4.6.5.2 Gradients for group velocity optimization

Frequency is not the only subject of dispersion optimization. Group velocity is also a common optimization objective [16]. Computing the design sensitivity of group velocity again requires eigenvector information at the wavevector of interest. It should be noted that traditional sensitivity computations for group velocities are more expensive than traditional sensitivity computations for frequencies because they require solving an extra system of equations. As with frequency optimization, there are two methods that we can use which will be advantageous in different situations.

First, we can use the surrogate model to infer the dispersion relation, and identify a wavevector of interest. Then we can solve the eigenvalue problem at that wavevector, and use Eqs. 4.6, 4.7 to compute the group velocity design sensitivities.

Alternatively, we can avoid group velocity sensitivity computations altogether by employing the chain rule. Using the surrogate model, the analytical expression for the predicted group velocity, Ψ_*, is

$$\Psi_* = g(\Gamma_*, \Gamma_R)\alpha \tag{4.19}$$

where g is the gradient of the covariance function with respect to its first input. Since the covariance function is fixed, this can be estimated numerically once, and stored for further use. Subsequently, we apply the chain rule to write the sensitivity of the predicted group velocity with respect to the design variables:

$$\frac{\partial \Psi_*}{\partial \theta} = \frac{\partial \Psi_*}{\partial \Omega_R} \frac{\partial \Omega_R}{\partial \theta} \tag{4.20}$$

where

$$\frac{\partial \Psi_*}{\partial \Omega_R} = g(\Gamma_*, \Gamma)(k(\Gamma, \Gamma) + \sigma_\omega^2 I)^{-1} \tag{4.21}$$

and $\frac{\partial \Omega_R}{\partial \theta}$ is again given by Eq. 4.18.

Using the second method requires traditional frequency sensitivity computations for a potentially large number of regression points. However, it gracefully avoids the need for any traditional group velocity sensitivity computations, which require solving a rather large linear system. This second method is advantageous to use when we desire to optimize group velocity at a large number of wavevector points.

4.7 Results

In this section, we apply our surrogate model to dispersion relations of two-dimensional materials and evaluate its performance against traditional methods and an existing deep-learning-based surrogate model.

4.7.1 Two-dimensional dispersion relations

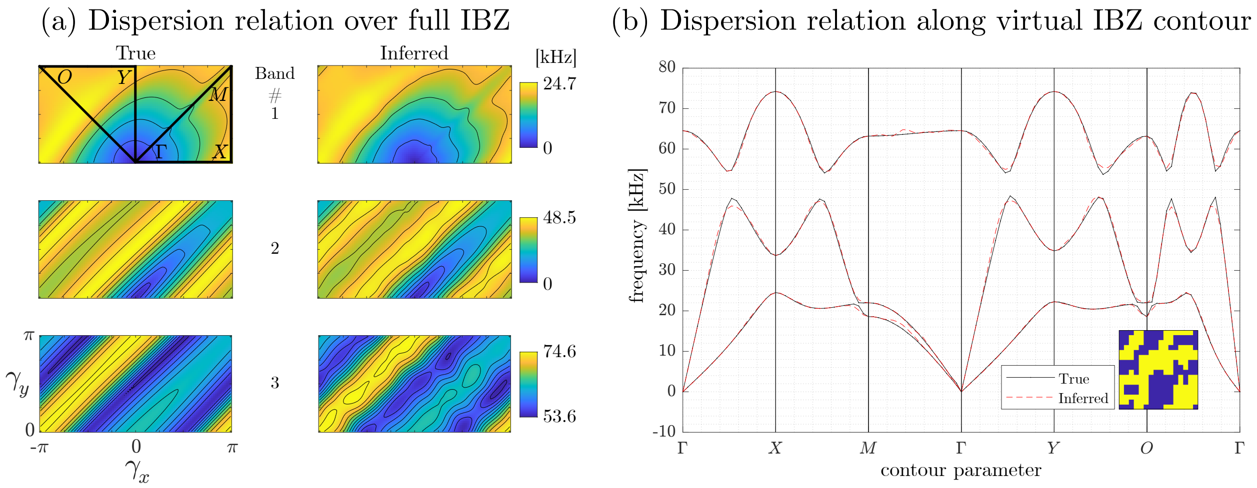
When modeling materials in two dimensions, the evaluation of the dispersion relation becomes more complicated and more computationally costly than in one dimension. This is for two reasons. First, a two-dimensional material requires more degrees of freedom to be accurately modeled. Second, wave propagation in a 2-D material must be described by a *wavevector*, rather than a *wavenumber*, since wave propagation may occur in any combination of the two directions. This means that one must sweep over a 2-D parameter space of wavevectors, solving an eigenvalue problem at each point in this 2-D space. These issues are further exacerbated when working with 3-D materials.

For this reason, when considering 2-D and 3-D materials, engineers and scientists often evaluate the dispersion relation only on the IBZ contour or a “virtual IBZ contour” rather than the full IBZ.

The definition of the IBZ contour is dependent on the symmetries of the unit cell, and is defined as the boundary of the IBZ for symmetry groups with reflection or glide-reflection symmetries. For symmetry groups without reflection or glide-reflection symmetries, the IBZ contour is not defined since the IBZ can be arbitrarily rotated, giving an infinite number of possible choices for the IBZ contour, and an infinite number of possible resulting dispersion band diagrams.

For example, Fig. 4.5 shows a unit cell lacking symmetry. Unit cells lacking symmetry belong to the $p1$ plane symmetry group (or *wallpaper group*). The IBZs of unit cells in the $p1$ symmetry group can be rotated arbitrarily, so a unique IBZ contour is not defined. As a substitute, the authors of [136] use a "virtual IBZ contour". Although the construction of this contour is entirely artificial, we include it because many people in the acoustics and phononics field are accustomed to seeing dispersion information in this format. Even for unit cells that have well-defined IBZ contours, the commonly used *contour shortcut* yields only partial information about wave propagation through the medium. The goal of our surrogate model is to increase the accessibility of dispersion information over the *full* IBZ.

In Fig. 4.5, we show the unit cell along with the first three bands of its dispersion relation evaluated along the "virtual IBZ contour", and over the full IBZ. We also show the inference of the surrogate model over these domains.


(a) Dispersion relation over full IBZ

(b) Dispersion relation along virtual IBZ contour

Figure 4.5: (a) The dispersion relation of a bimaterial unit cell evaluated over the full IBZ. The true dispersion band is shown in the left column, and the inference given by the surrogate model is shown in the right column. Here, the surrogate model uses a training set of 2000 dispersion relations to construct the covariance function, and uses 50 regression points (regression points not shown). (b) The same dispersion relation evaluated along the "virtual IBZ contour", $\Gamma XM\Gamma YO\Gamma$. The unit cell design is shown in the inset, where yellow represents a stiff and heavy material ($E = 200\text{GPa}$, $\rho = 8000\text{kg/m}^3$) and blue represents a softer and less dense material ($E = 200\text{MPa}$, $\rho = 800\text{kg/m}^3$). The side length of the unit cell is $a = .01\text{m}$. This unit cell has no symmetry, making it part of symmetry group $p1$ [136]. The IBZ of this symmetry group can be arbitrarily rotated, so a well-defined IBZ contour does not exist [136]. However, since so many are accustomed to visualizing dispersion information in this format, we've included the dispersion relation evaluated on the "virtual IBZ contour" as defined in [136]. Here, we again show the true dispersion relation along with the inferred dispersion relation.

4.7.2 Generating random designs

In this section, we consider 2D unit cell designs of the $p1$ symmetry group (i.e. no symmetry) using a plane stress model. To generate random unit cell designs, we sample a Gaussian process with a 2-D periodic covariance function. To construct this covariance function, we start with a 1-D periodic covariance function,

$$k_{\text{per-1D}}(x, x') = \sigma_f^2 \exp\left(-\frac{2}{\ell^2}\sin^2\left(\pi\frac{|x - x'|}{p}\right)\right) \tag{4.22}$$

where σ_f controls the overall variance of the Gaussian process, ℓ controls the lengthscale of the randomness, and p is the period. In our case, we set p to be the side length, a, of the unit cell. We form the 2-D periodic covariance function as the product of the 1-D periodic covariance function applied to each direction:

$$k_{\text{per-2D}}(x, y, x', y') = k_{\text{per-1D}}(x, x')\, k_{\text{per-1D}}(y, y') \tag{4.23}$$

where x and y are the horizontal and vertical coordinates of locations in the unit cell. We emphasize that the Gaussian process used to generate unit cell designs has no direct connection with the Gaussian process used by the surrogate model for inference.

Beyond the periodic covariance parameters (which control the variance and the lengthscale of features in the unit cell), there are several other parameters that define the unit cell. The lattice parameter a dictates the side length of the square unit cell. The resolution of the unit cell is determined by the integer N_{pix}, which defines the number of pixels both vertically and horizontally. The constitutive properties of each pixel are bounded by E_{min}, E_{max}, ρ_{min}, ρ_{max}, ν_{min}, and ν_{max}, corresponding to elastic modulus, mass density, and Poisson's ratio respectively. The number of unique materials in each design is controlled by N_{mat} (for example, a bimaterial unit cell would be defined by $N_{mat} = 2$). Further details of unit cell design generation are given in 4.10.1. Examples of how these unit cells might look are provided in section 4.7.6.

4.7.3 Visualization of the acquisition function for the two-dimensional problem

In this section, we visualize the covariance and acquisition functions for the two-dimensional problem. In Fig. 4.6, we show the evolution of the acquisition function and the resulting variance fields for a covariance function constructed from a dataset of 2000 dispersion relations.[1] As we compute the order of regression point sampling, we can observe that points that are not near to each other can be highly covariant. For example, the covariance function is taking advantage of the periodicity of the dispersion relation. In the example shown in Fig. 4.6, the unit cells in the dataset used to construct the covariance function belong to the *p1* plane symmetry group (they have no symmetry), which is a superset of all other plane symmetry groups. Due to the construction of this symmetry group's IBZ and the periodicity of the dispersion relation, the solution along the boundary of the IBZ must be symmetric across the line $\gamma_x = 0$. Indeed, we see from the $(1, 1)$ and $(2, 1)$ panels that a single regression point on the far left side of the IBZ causes a "cold spot" in the variance field on the right side of the IBZ in the region that is "close" to the regression point when considering the periodicity of the dispersion surface. This indicates that the model gains insight about the function value on the right side of the domain from information observed on the left. Our covariance function is able to implicitly learn crucial patterns such as periodicity and symmetry from the data, without the need for a user to manually encode these assumptions.

---

[1] The dataset was constructed with the following parameters: $\sigma_f = 1$, $\ell = 0.5$, $a = 1\,\mathrm{m}$, $N_{pix} = 16$, $N_{mat} = 2$, $E_{min} = 200\,\mathrm{MPa}$, $E_{max} = 200\,\mathrm{GPa}$, $\rho_{min} = 800\,\mathrm{kg/m^3}$, $\rho_{max} = 8000\,\mathrm{kg/m^3}$, $\nu_{min} = 0$, $\nu_{max} = 0.5$. The wavevector resolution of each dispersion relation in the dataset is 101×51.

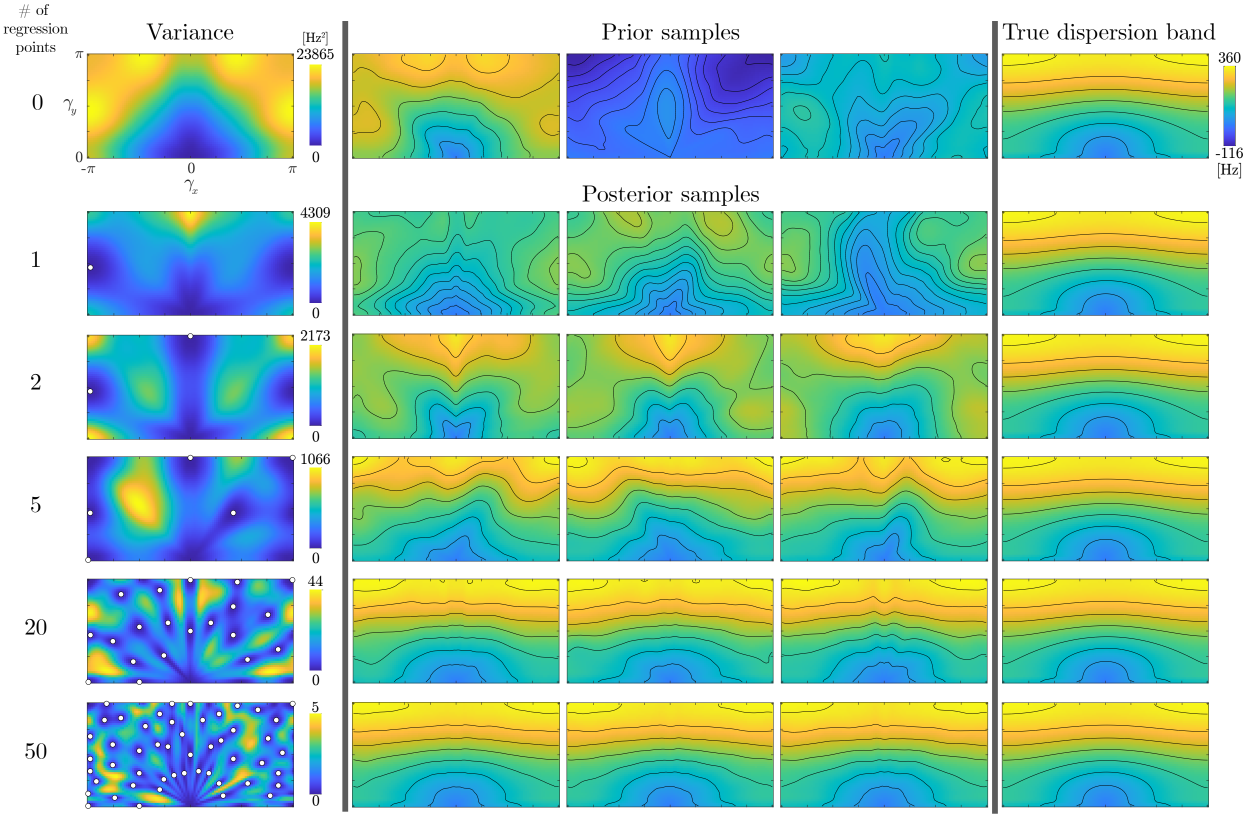


Figure 4.6: Left column: from top to bottom, we show the progression of regression point selection for 0, 1, 2, 5, 20, 50 regression points. Right column: the true dispersion relation for one particular unit cell design. Each image in this column is identical. Center three columns: three random samples drawn from the posterior distribution for 0, 1, 2, 5, 20, 50 regression points. Note that when the model uses 0 regression points, the posterior distribution is identical to the prior distribution. Also, we observe that as more regression points are used, the samples become more similar to each other, and to the true dispersion relation in the right column. Further, note that the posterior mean function is the final hypothesis of the surrogate model, not a randomly drawn posterior sample.

4.7.4 Inference examples

We now show examples of our surrogate model infering the dispersion relation for two-dimensional structures using a sparse set of regression points in Fig. 4.7. In each case, the model uses a dataset of 2000 dispersion relations (the same dataset used to generate Fig. 4.6) and 50 regression points (the same regression points shown in the bottom left panel of 4.6). The dataset was constructed with the following parameters: $\sigma_f = 1$, $\ell = 0.5$, $a = 1\,\text{m}$, $N_{pix} = 16$, $N_{mat} = 2$, $E_{min} = 200\,\text{MPa}$, $E_{max} = 200\,\text{GPa}$, $\rho_{min} = 800\,\text{kg/m}^3$, $\rho_{max} = 8000\,\text{kg/m}^3$, $\nu_{min} = 0$, $\nu_{max} = 0.5$. The wavevector resolution of each dispersion relation in the dataset is 101×51. In Fig. 4.7 we use 50 regression points to achieve a relatively accurate prediction; however, in practice, the number of regression points can be adjusted fluidly to strike the right balance of accuracy versus computational cost depending on the application.

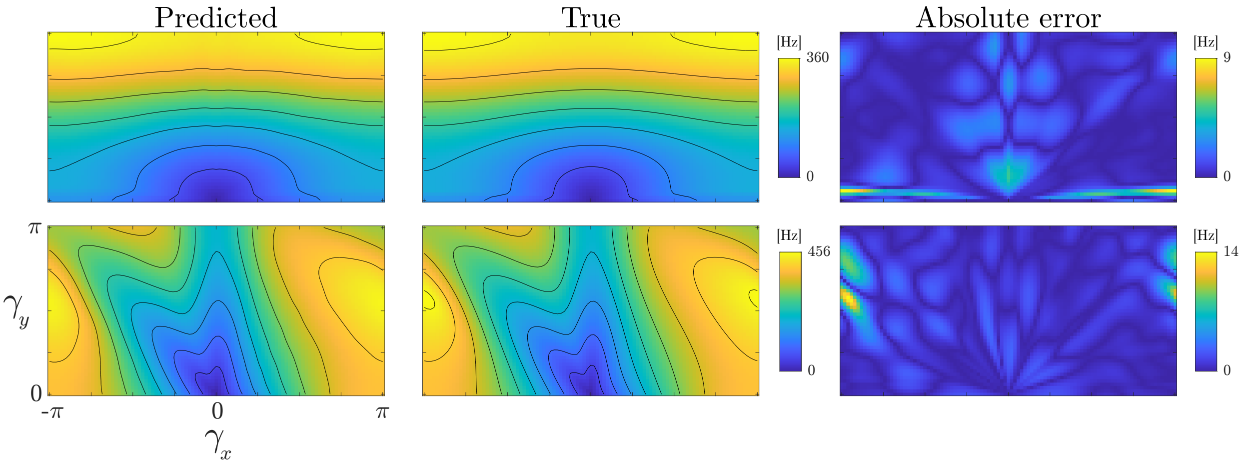


Figure 4.7: A comparison of the prediction against the truth for the first eigenvalue band of two different unit cells over the full IBZ. In each case, the surrogate model uses a covariance function trained on 2000 dispersion relations and 50 regression points. The absolute error is shown on the right for each scenario.

4.7.5 Eigenanalysis of the covariance function

In this section, to examine the relationship between the size of the covariance function training set and the expressivity of the model, we examine the spectra of covariance functions based on datasets of different sizes. Recall from section 4.5.2 that the number of nonzero eigenvalues of the covariance function corresponds to the size of the covariance function's eigenbasis, which is the basis the model uses to fit to the regression points. A covariance function with a more comprehensive eigenbasis results in a model with more expressive power, and a model that can make use of a larger number of regression points.

For a range of logarithmically spaced dataset sizes, we compute the spectrum of the corresponding covariance function, and plot the spectra in Fig. 4.8. As the size of the dataset increases, the number of nonzero eigenvalues increases, meaning that the covariance function's eigenbasis is expanding to be more expressive. Eventually, however, the addition of more dispersion relations to the dataset has a decreasing effect on the spectrum of the covariance function. For the largest dataset size (20,000 dispersion relations) we plot the eigenfunctions corresponding to the 10 largest eigenvalues of the covariance function in Fig. 4.9. These eigenfunctions are the first ten elements of the eigenbasis, and can be thought of as the ten most prominent and fundamental function shapes that the model uses to fit a dispersion relation.

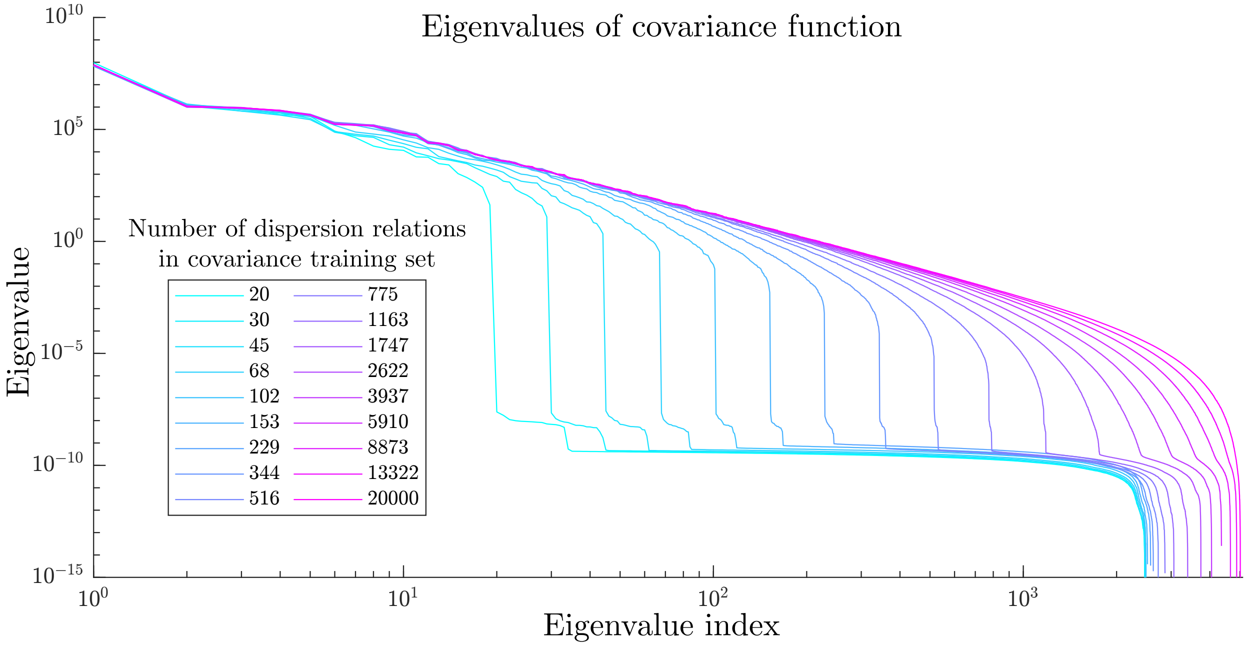


Figure 4.8: The spectra of covariance matrices computed from datasets containing different numbers of dispersion relations. The wavevector resolution of the datasets is 101×51, and the unit cell designs are 16×16 pixels. Because we plot in log scale, eigenvalues that are zero are not plotted. As expected from basic statistics, we observe that datasets with larger numbers of dispersion relations produce covariance matrices with more nonzero eigenvalues. This indicates that covariance functions generated from larger datasets will have larger eigenbases, and therefore have more expressive power. However, we observe that the change in spectrum appears to slow down at some point, and we get diminishing returns after about 1700 dispersion relations. We can expect this "saturation" effect to happen at different points in different scenarios, and will depend on many factors including the resolution of the dispersion relations in the datasets and the method of generating unit cell designs.

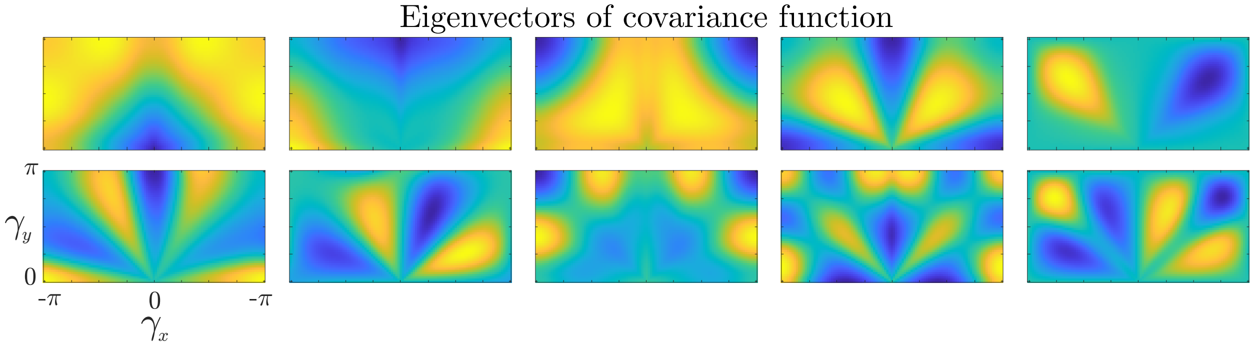


Figure 4.9: The first 10 eigenfunctions of the covariance function that uses training set containing 20,000 dispersion relations. The eigenfunctions form a basis for the hypothesis space of the Gaussian process regressor.

The expressivity of the model (i.e. the broadness of the hypothesis space) depends on the dataset used to construct the covariance function. In general, larger datasets will allow for more expressivity. This is not surprising, since other data-driven methods such as neural networks require larger amounts of data to characterize more complex functions. However, there is a difference in how the model complexity is determined. For a neural network, the hypothesis space is determined by the network architecture, which is typically determined before training the model. If too little training data is supplied, the hypothesis space of the network remains large, but the model's performance will be low since the training data was not sufficient to narrow down the hypothesis space. In contrast, the expressivity of our model's data-driven covariance function is determined by the dataset itself. With our method of covariance function construction, a small dataset will automatically result in a smaller hypothesis space, and a larger dataset will automatically result in a larger hypothesis space, without the need to adjust complicated model hyperparameters (such as network architecture).

4.7.6 Model evaluation

In the following subsections, we compare the performance of the GPR model against two other models in the context of the two-dimensional problem. The first model is linear interpolation, representing the traditional model for computing dispersion relations.[2] The second model is an existing convolutional neural network surrogate model from the literature [84].

4.7.6.1 The datasets

To evaluate the performance of the GPR model, we consider the accuracy of the model as a function of the size of the covariance training set and the number of regression points for both in-distribution and out-of-distribution scenarios. We generate three datasets, one for training the covariance function, one for testing in-distribution performance, and one for testing out-of-distribution performance. The training dataset contains dispersion relations of 5000 unit cells generated as described in section 4.7.2, with periodic kernel parameters $\sigma_f = 1$ and $l = 0.5$ at a resolution of 32×32 ($N_{pix} = 32$). The unit cells have a lattice parameter $a = 1\,\mathrm{m}$, are divided into 2 materials ($N_{mat} = 2$), and have material properties $E_{min} = 200\,\mathrm{MPa}$, $E_{max} = 200\,\mathrm{GPa}$, $\rho_{min} = 800\,\mathrm{kg/m^3}$, $\rho_{max} = 8000\,\mathrm{kg/m^3}$, $\nu_{min} = 0$, and $\nu_{max} = 0.5$. The in-distribution test dataset contains dispersion relations of 100 unit cells randomly generated using the same parameters as the training set (hence, *in-distribution*). The out-of-distribution test dataset contains dispersion relations of 100 unit cells randomly generated with the same parameters as the training set, except $l = 1$, $N_{mat} = 3$, $E_{min} = 200\,\mathrm{GPa}$, $E_{max} = 200\,\mathrm{TPa}$. These four parameters cause the distributional shift between the training set and the out-of-distribution test set. Although the elastic moduli in the out-of-distribution dataset are outside of the range of realistic materials, we use them to show

---

[2] For the traditional linear interpolation model, one simply evaluates the dispersion relation on a grid of evenly spaced points and linearly interpolates if evaluation is needed at points that do not belong to the grid.

that our model works when the material properties and frequency ranges of the training set and test set are vastly different. The training set uses a wavevector resolution of 101×51, and the test sets use a wavevector resolution of 501×251. Examples of unit cells drawn from each distribution are shown in Fig. 4.10.

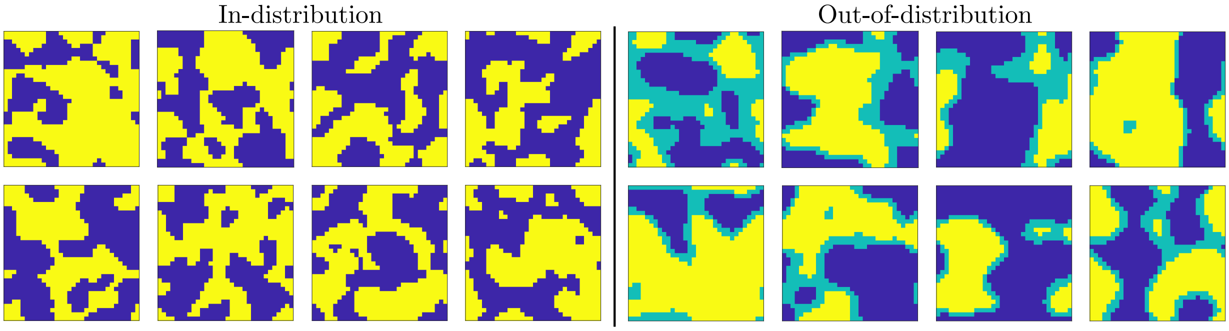


Figure 4.10: A visualization of the unit cell designs for the two distributions used to generate the datasets. Blue represents 0, teal represents 0.5, and yellow represents 1. On the left we show in-distribution designs, generated with the same parameters as the training set. On the right, we show out-of-distribution designs, generated by parameters different from the training set. The out-of-distribution dataset is generated with a larger length scale parameter, an extra material, and elastic moduli 1000× larger (note that the colorscale does not show this stiffness increase).

4.7.6.2 Performance metrics

To measure the error of the model, we use the L^2 norm and the H^1 norm, defined in Eqs. 4.24, 4.25. The L^2 norm allows us to assess how well the model infers frequency values, and the H^1 norm to assess how well the model infers group velocity values. Errors are evaluated at the resolution of the test sets, which are about five times more resolved along each of the two wavevector directions, as described in section 4.7.6.1. After evaluating model error on each eigenvalue band of each dispersion relation in the test set, we average over eigenvalue bands, and take the 0.95 quantile over dispersion relations to get a sense for worst-case error. The norms are given as

$$\|f\|_{L^p}^p = \int_D \|f(x)\|^p \, \mathrm{d}x \tag{4.24}$$

$$\|f\|_{H^k} = \left(\sum_{0 \le |\alpha| \le k} \|\partial^\alpha f\|_{L^2}^2 \right)^{1/2} \tag{4.25}$$

where ∂^α denotes the multi-index partial differential operator, $|\alpha| = \sum_i \alpha_i$, and D is some domain over which we are interesting in measuring the function. In our case, D will be the IBZ, and the function f will be the error of the model.

4.7.6.3 Comparison against linear interpolation model

To compare against the linear interpolation model, we consider the number of regression points used by each model and the error that each model incurs in its prediction. In this subsection, because of the 1000× shift in elastic modulus of the out-of-distribution test set, we divide the out-of-distribution errors by $\sqrt{1000}$ so that they can be readily compared to the in-distribution errors.[3] We find that the GPR model always uses fewer regression points than the corresponding linear interpolation model achieving the same accuracy. At most, in this particular study, the GPR model uses 7.75 times fewer regression points than the equivalent linear interpolation model. Because almost all of the computational cost is due to regression point eigenvalue computations, this can be seen as a speed up of approximately the same factor. The results are shown below in Fig. 4.11.

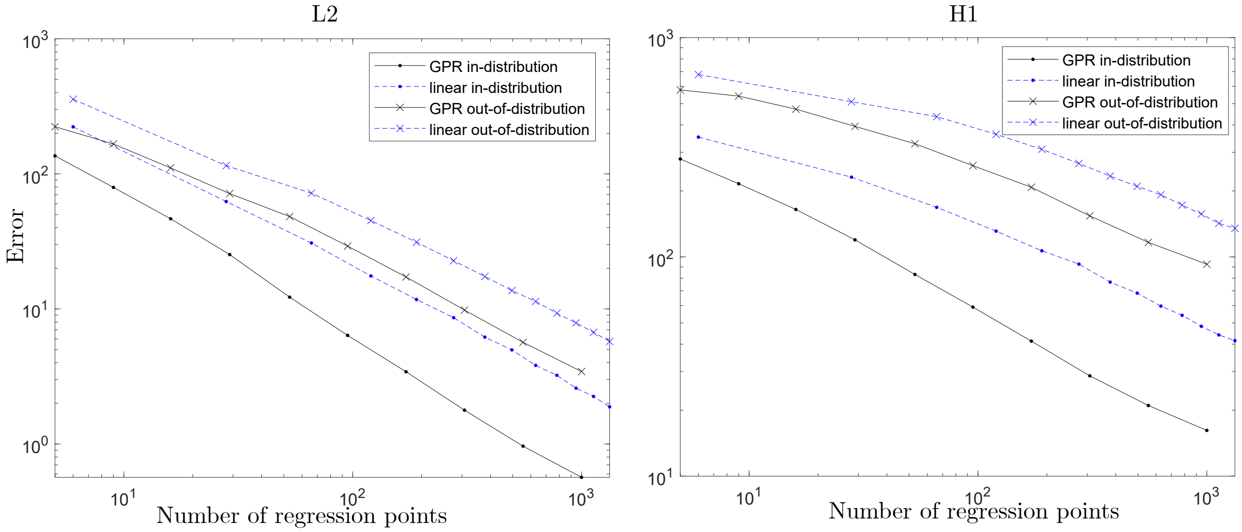


Figure 4.11: A comparison of the GPR model against the traditional linear interpolation model. Performance is evaluated on two datasets: in-distribution and out-of-distribution. In both cases, the GPR model performs substantially better than the linear interpolation model. In the H^1 plot for in-distribution performance, the GPR model is shown to achieve the same error as the linear interpolation model while using up to 7.75× fewer regression points.

4.7.6.4 Comparison against existing CNN surrogate model

We compare our model performance to the performance of the CNN surrogate model proposed by Finol et al. [84]. Please refer to 4.10.2 for details about the CNN architecture and training. Note that the CNN uses the material properties of the unit cell in the input array. This is in contrast to

---

[3] This exactly transforms the errors to what their values would be if the modulus range were 1000× smaller, since the dispersion relation of a unit cell whose modulus is increased by 1000× will experience an upshift in frequency at a factor of $\sqrt{1000}$, and because both the linear interpolation and GPR models' predictions (and therefore errors) are linearly proportional to the frequency values of the regression points.

the GPR model, which does not directly observe the material properties of the unit cell, but instead observes regression points of the dispersion relation.

In this subsection, we allow the GPR models to observe only 50 regression points, and limit both the CNN and GPR models training set sizes to range from only 50 to 5000 dispersion relations. For each training set size, we train the CNN for a maximum of 30 epochs with early stopping based on a validation set. Training the CNN on the training set of 5000 dispersion relations takes approximately 24 minutes/epoch on an NVIDIA GeForce RTX 3080 Ti GPU. For each training set size, we also compute a new covariance function for the GPR, taking only 1 second for the largest training set size (5000). Because training a neural network depends on the randomly initialized weights of the network, we repeat the training 10 times with different random initializations to get a sense for the variation in results. After measuring the L^2 and H^1 norms of the error, we average over all 10 eigenvalue bands, and take the 95% quantile error over unit cells. Additionally, for the CNN, we measure the standard deviation of this error over the ten trials and plot the ± one standard deviation spread.

The results (Fig. 4.12) show that the GPR model achieves errors significantly lower than the CNN for all training set sizes. This is because we are using relatively small training sets, and neural networks typically require more data to train effectively. For reference, Finol et al. used a minimum of 20,000 and up to 100,000 dispersion relations to train their CNN. In general, it is a nontrivial task to adjust the architecture (and therefore expressivity) of a neural network to be appropriate for the amount of training data that you have. This process generally involves a substantial amount of trial-and-error and hyperparameter tuning, which can be costly in terms of time and effort. Note that in this section, we do not adjust the out-of-distribution errors by the factor of $\sqrt{1000}$ because this transformation would not be valid for the CNN, so in-distribution and out-of-distribution results cannot be directly compared against each other. For both in-distribution and out-of-distribution, we see the CNN gradually improve with training set size in the L^2 norm, but not in the H^1 norm. This makes sense, because the loss function that the CNN uses (mean squared error) is not sensitive to mismatches in gradients (H^1 error).

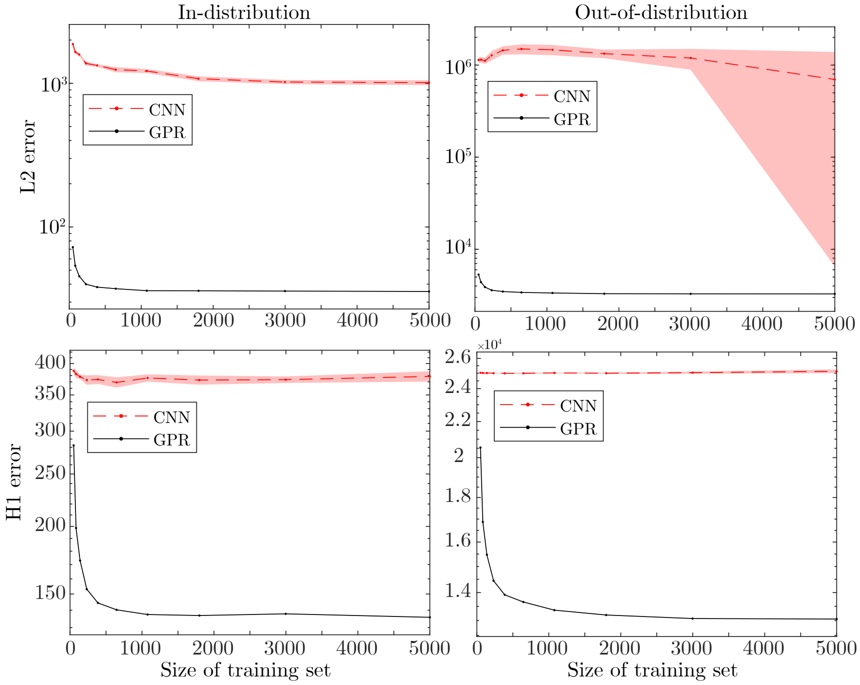


Figure 4.12: Comparison of GPR surrogate model against CNN surrogate model for in-distribution and out-of-distribution performance in L^2 and H^1 norms. The GPR model performs with much lower error than the CNN for these relatively small training set sizes.

4.7.7 Three-dimensional performance evaluation

This section is not included in the version of the manuscript as published on *Computer Methods in Applied Mechanics and Engineering*. Here, we demonstrate an extension of the surrogate model to three-dimensional material systems, showing a drastic increase in benefit as compared to two-dimensional setting.

In this section, only out-of-distribution performance is evaluated. Examples of in-distribution (training) and out-of-distribution (test) unit cells used to generate the dispersion relations in the training sets are shown below in Fig. 4.13. The unit cell designs of both datasets are defined on a $10 \times 10 \times 10$ voxel grid. The in-distribution dataset unit cell designs allow an infinite number of constituent materials (material property values are not rounded to some finite set of values), while the out-of-distribution dataset adheres to a bimaterial design paradigm. The in-distribution dataset unit cells also uses a longer feature lengthscale ($l = 1$) than the out-of-distribution dataset ($l = 0.5$). Lastly, the dispersion relations of the in-distribution (training) dataset are computed at a resolution of $11 \times 11 \times 6$, whereas those of the out-of-distribution (test) dataset are computed at a resolution of $21 \times 21 \times 11$. The out-of-distribution (test) set contains 100 dispersion relations. Two different sizes of training sets are used. The smaller training set contains 234 dispersion relations, and the larger training set uses 1322 dispersion relations. The model exhibits larger efficiency gains than demonstrated in the two-dimensional setting. Recall that the improvement observed in the two-dimensional setting was approximately a 7.75× speed-up factor in the best case for *in-distribution* performance (whereas we are now only measuring *out-of-distribution performance*), with a training set of 5000 dispersion relations. In the results shown (Fig. 4.14), the model achieves an impressive out-of-distribution performance of a 20× speed-up as compared to traditional linear interpolation models, with substantially fewer dispersion relation examples in the training sets than in the two-dimensional case. These results in particular show potential for the use of this model in the engineering of real three-dimensional materials.

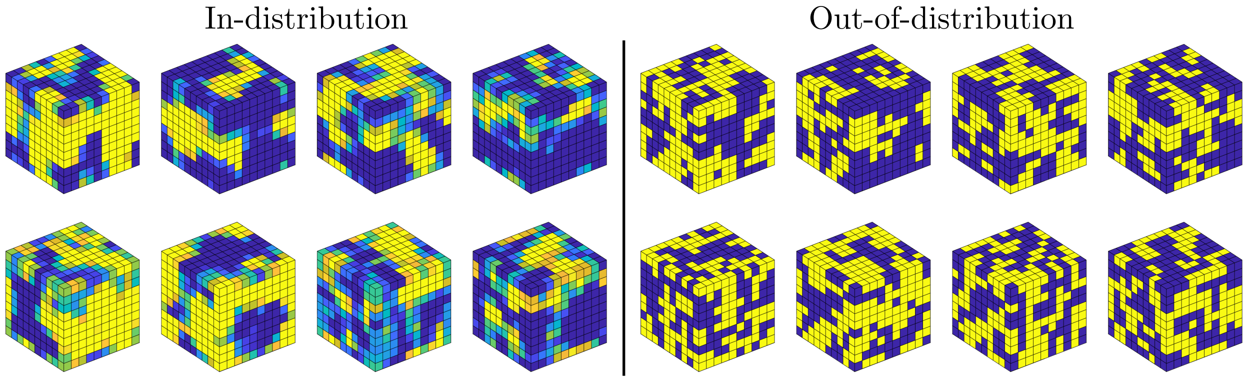


Figure 4.13: Examples of in-distribution and out-of-distribution unit cells for the evaluation of the GPR-dispersion model in a three-dimensional setting.

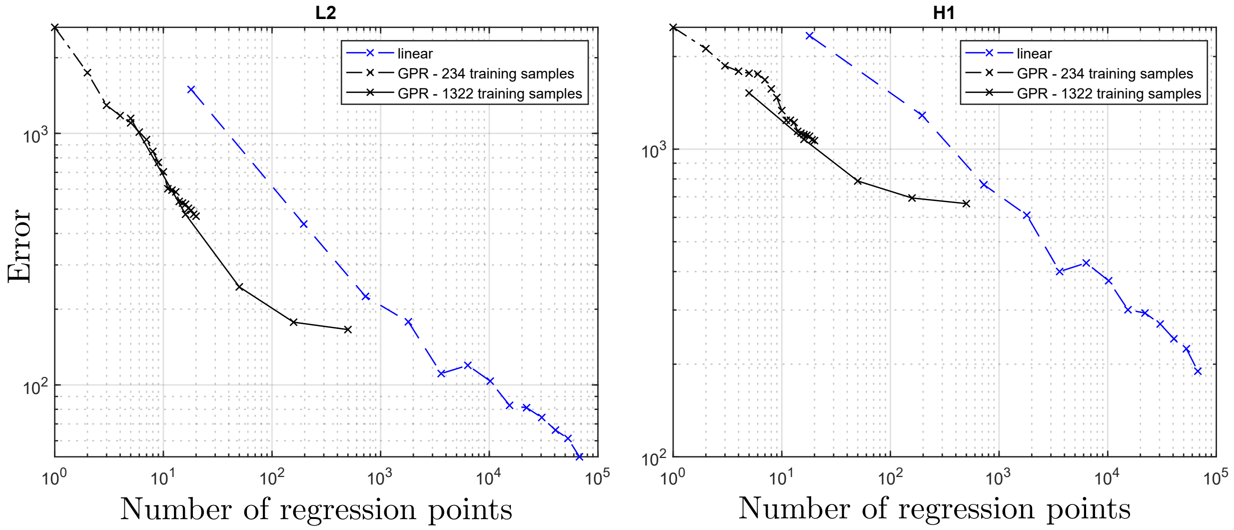


Figure 4.14: A comparison of the GPR-dispersion surrogate model against linear interpolation.

4.8 Discussion

Interest in the design of multi-functional wave devices and the high computational cost of dispersion relation computations motivate the need for more efficient methods of dispersion relation evaluation, including surrogate models. We have introduced a new GPR-based surrogate model for the accelerated computation of dispersion relations over the full irreducible Brillouin zone.

The GPR model builds its covariance function from data and is able to learn from datasets that are 2-3 orders of magnitude smaller than datasets required by the demonstrated deep learning approaches to date. Instead of involving a typical deep learning training process, the model uses only a simple covariance calculation to capture the data trends. The low requirement for data, along with the avoidance of a typical training process, massively reduces the overhead cost to set up the model, allowing the model to be more easily repurposed for new design tasks.

Because the model never directly observes the parameters governing the unit cell design, it can seamlessly train on unit cells represented in one format (e.g. pixelated/voxelated) and subsequently generalize its predictions to unit cells of other representations (e.g. parametric). This inherent quality makes the model versatile, emphasizing its potential to be effectively applied across a wide spectrum of design scenarios.

Finally, the model can supply analytically computed gradients for use with gradient-based design optimization, along with uncertainty estimates - both highly useful tools for design tasks. For optimization problems where deviation from the solution of the variational method cannot be tolerated, the surrogate model can simply be used for the first several iterations of optimization and then handed-off to a slower, but more exact dispersion evaluation method. Further, the model can work in tandem with existing, highly specialized variational methods for dispersion computations. We hope that by increasing the computational accessibility of full dispersion relations we will alleviate the need for current wave analysis shortcuts and design simplifications, and enable the design of next-generation, highly multi-functional wave devices.

A current limitation to our surrogate model is that the covariance function treats each eigenvalue band separately, rather than considering information from all bands together to make more informed predictions. Not only does this likely reduce the accuracy-per-regression-point of the model, but it also means that the order in which regression points are acquired by the model is different for each eigenvalue band, which is inconvenient for computations where multiple bands are to be considered simultaneously. As a first order improvement, this could be explored by simply expanding the covariance function to consider the covariance between eigenfrequencies in different bands. However, if this method is taken to the extreme, the covariance matrix that the covariance function interpolates from can quickly become large enough to be a concern for memory requirements.[4] For

---

[4] For example, if a dataset of dispersion relations has a resolution in reciprocal space of 201×201, the covariance

this reason, it may be important to examine more efficient ways of storing covariance information. For example, it may be reasonable to only consider covariance information between bands that are within N bands of each other. Another alternative could be to characterize the covariance function in a compressed representation as a neural network rather than an explicit interpolation function.

Further, integration of this GPR model with existing physics-based techniques [137–144] has the potential to enable the hyper-efficient computation of dispersion relations. This could be done simply by using these techniques to carry out regression point computations within the GPR model, or even to generate the training sets that are used to inform the covariance function.

Other future work may involve determining more optimal patterns for regression point sampling. Although the current method works well, it is greedy, and more optimal acquisition functions likely exist. For example, one may consider at each iteration selecting the point that most reduces the variance field in some L_p norm, or may even consider setting up an optimization problem to jointly determine the optimal locations of a fixed number of regression points.

These improvements may improve the practical applicability of this surrogate model and may further decrease the computational barrier to accessing full dispersion information, enabling exciting advances in the design of wave devices.

4.9 Acknowledgements & Code

A.C.O. is funded by DOE award no. DE-SC0021253. B.T.F. is supported by an NSF GRFP Fellowship and a Kortschak Scholarship.

Code generated during this project for computing dispersion relations of pixelated structures using the finite element method, and for using the GPR-dispersion model can be found on github at the first two links below. Analogous codes for computation of dispersion relations in three-dimensions can be found at the third and fourth links below.

- https://github.com/aco8ogren/2D-dispersion
- https://github.com/aco8ogren/2D-dispersion-GPR
- https://github.com/aco8ogren/3D-dispersion-matlab
- https://github.com/aco8ogren/3D-dispersion-GPR

---

matrix is of size 40401 × 40401, occupying 13.06GB of memory when stored in double precision. This means that if the covariance between even two bands at this resolution is considered, the resulting covariance matrix will be 80802 × 80802, occupying 52.24GB of memory.

4.10 Supplementary information

The following sections give supplementary information for this chapter, including details on how random designs were generated, the architecture and training of the CNN that GPR-dispersion was compared to, and a mathematical derivation of the gradients used for design optimization of dispersion relations in the context of the finite element method.

4.10.1 Generation of random designs

One dimension

We generate random 1-D unit cell designs with material properties that vary smoothly except at interfaces where the material properties are allowed to be discontinuous. To accomplish this, we draw the exponents of the material properties from a Gaussian process (distinct from the Gaussian process used for regression) using a Matérn covariance function modified to have block-diagonal support. The Matérn covariance structure allows us to impose smoothness semiglobally over the unit cell, while the block diagonal support allows for the interfaces of discontinuity in material property values.

Two dimensions

To generate a random design, we draw a random function from the constitutive Gaussian process (described in section 4.7.2) at the resolution determined by N_{pix}. We use a mean function of $m(x) = 0.5$ for the constitutive Gaussian process. Any values below 0 or above 1 are trimmed to 0 and 1 respectively. Finally, each pixel's value is rounded to the nearest of N_{mat} equally spaced values between 0 and 1. The material properties of each pixel are then linearly interpolated between the maximum and minimum constitutive property values.

4.10.2 CNN description

Recall that each unit cell is 32×32 pixels, and each pixel has its own elastic modulus, density, and poisson ratio. Therefore each unit cell can be represented by a constitutive array, A_{const}, of size $32 \times 32 \times 3$. To form the CNN input array, we concatenate A_{const} with two more matrices, γ_x and γ_y, which are matrices of size 32×32 with all elements equal to the x and y wavevector components respectively. The input array, X_{CNN}, has size $32 \times 32 \times 5$. The output array, y_{CNN}, is a vector of length 10 containing the 10 lowest eigenfrequencies corresponding to the given unit cell and wavevector. For model architecture, we borrow from Finol et al. [84]. The model architecture is as follows: an input layer matching the size of the input, followed by two Conv2D ReLU layers with 275 filters of size 2×2, followed by a MaxPooling2D layer of size 2×2, followed by a flattening layer (to transition to fully connected layers), followed by two fully connected ReLU layers each with 500 nodes, finally followed by the output layer of size 10.

4.10.3 Derivation of gradients for optimization of dispersion relations

In order to optimize the dynamic properties of material designs, we need to be able to efficiently compute the gradient of those properties with respect to design variables. To do this, we start with the generalized eigenvalue equation used for dispersion analysis of periodic materials.

$$(K(\gamma) - \mu M)u = 0 \tag{4.26}$$

Note that, due to the hermitian symmetry of K and M, the following is also true (and will be used later).

$$u^*(K(\gamma) - \mu M) = 0 \tag{4.27}$$

where * denotes conjugate transpose.

By taking a gradient with respect to a generic variable θ, we obtain Eq. 4.28. For now, consider θ to be a scalar. To generalize to vector θ (possibly as a design vector or a wavevector), simply repeat these computations for each entry of θ. The product rule is used here.

$$\frac{\partial K}{\partial \theta}u - \frac{\partial \mu}{\partial \theta}Mu - \mu\frac{\partial M}{\partial \theta}u + (K - \mu M)\frac{\partial u}{\partial \theta} = 0 \tag{4.28}$$

There are two things that can be done with this equation. First, if we only want the gradient of the eigenvalue with respect to a design vector, we can left-multiply Eq. 4.28 by u^*. Then, applying Eq. 4.27, we can see that

$$\frac{\partial \mu}{\partial \theta} = u^*(\frac{\partial K}{\partial \theta} - \mu\frac{\partial M}{\partial \theta})u \tag{4.29}$$

Alternatively, if the gradient of the eigenvector u is also desired, Eq. 4.28 can be rearranged to

$$(K - \mu M)\frac{\partial u}{\partial \theta} - \frac{\partial \mu}{\partial \theta}Mu = -(\frac{\partial K}{\partial \theta} - \mu\frac{\partial M}{\partial \theta})u \tag{4.30}$$

If $\theta \in \mathbb{R}$ and $K, M \in \mathbb{R}^n$, then this system has n equations but $2n$ unknowns (or in general if $\theta \in \mathbb{R}^d$, and $K, M \in \mathbb{R}^n$, then this system has nd equations but $n(d+1)$ unknowns). This is because eigenvectors are only determined up to their scale. To provide the extra n equations required to solve, we can use an eigenvector normalization equation. It's standard to normalize eigenvectors by their mass matrix.

$$u^*Mu = 1 \tag{4.31}$$

By differentiating, we arrive at

$$\frac{\partial(u^*)}{\partial\theta}Mu + u^*\frac{\partial M}{\partial\theta}u + u^*M\frac{\partial u}{\partial\theta} = 0$$

$$(\frac{\partial u}{\partial\theta})^*Mu + u^*\frac{\partial M}{\partial\theta}u + u^*M\frac{\partial u}{\partial\theta} = 0$$

$$(\frac{\partial u}{\partial\theta})^*Mu + u^*M\frac{\partial u}{\partial\theta} = -u^*\frac{\partial M}{\partial\theta}u \tag{4.32}$$

If our eigenvector u were purely real, we could say:

$$(\frac{\partial u}{\partial\theta})^TMu = ((\frac{\partial u}{\partial\theta})^TMu)^T = u^TM\frac{\partial u}{\partial\theta}$$

This would make Eq. 4.32 easy to rewrite so that the unknowns on the LHS only receive operations from the left.

Then we could solve

$$\begin{bmatrix} K-\mu M & -Mu \\ & \\ -u^TM & 0 \end{bmatrix} \begin{Bmatrix} \frac{\partial u}{\partial\theta} \\ \frac{\partial\mu}{\partial\theta} \end{Bmatrix} = \begin{Bmatrix} -\left(\frac{\partial K}{\partial\theta} - \mu\frac{\partial M}{\partial\theta}\right)u \\ \frac{1}{2}u^T\frac{\partial M}{\partial\theta}u \end{Bmatrix} \tag{4.33}$$

However, in this case, our Bloch-Floquet periodic boundary conditions cause our eigenvector u to be complex, so in general

$$(\frac{\partial u}{\partial\theta})^*Mu \neq ((\frac{\partial u}{\partial\theta})^*Mu)^* = u^*M\frac{\partial u}{\partial\theta} \tag{4.34}$$

and therefore

$$(\frac{\partial u}{\partial\theta})^*Mu = (u^*M\frac{\partial u}{\partial\theta})^* \tag{4.35}$$

By substituting Eq. 4.35 into Eq. 4.32, we can arrive at

$$(u^*M\frac{\partial u}{\partial\theta})^* + u^*M\frac{\partial u}{\partial\theta} = -u^*\frac{\partial M}{\partial\theta}u \tag{4.36}$$

In other words,

$$2\,\mathrm{Re}\left[u^{*}M\frac{\partial u}{\partial\theta}\right]=-u^{*}\frac{\partial M}{\partial\theta}u \tag{4.37}$$

And there is no constraint on the imaginary component. In other words,

$$\mathrm{Im}\left[u^{*}M\frac{\partial u}{\partial\theta}\right]=\text{anything} \tag{4.38}$$

To solve this, I can break up the imaginary and real equations, and write them individually. This turns 3 complex equations into 5 real equations. The reason I don't obtain 6 real equations from the 3 complex equations (as one may expect) is that Eq. 4.36 actually only provides one real equation (note that of Eqs. 4.37, 4.38, only Eq. 4.37 tells us something useful). Luckily, because our eigenvectors are complex, they not only need a constraint in magnitude, but also in complex angle. Note that if u is an eigenvector, then $uRe^{i\beta}$ is also an eigenvector for any $R,\beta\in\mathbb{R}$. In other words, the eigenvector can be multiplied by any complex scalar, and still be an eigenvector. For this reason, we not only need to constrain the magnitude of u (as we have done with Eq. 4.31), we also need to constrain the complex angle of the entries of u. To do this, we can simply enforce that the first entry of u is purely real. This can be written as

$$\mathrm{Im}\left[\frac{\partial u}{\partial\theta}\right]=0 \tag{4.39}$$

Finally, we convert Eq. 4.37 into a form conducive to being encoded into a matrix by noting that

$$\mathrm{Re}\left[u^{*}M\frac{\partial u}{\partial\theta}\right]=\mathrm{Re}\left[u^{*}M\right]\mathrm{Re}\left[\frac{\partial u}{\partial\theta}\right]-\mathrm{Im}\left[u^{*}M\right]\mathrm{Im}\left[\frac{\partial u}{\partial\theta}\right] \tag{4.40}$$

Finally, we can combine Eqs. 4.30, 4.37, 4.39 to write the following matrix equation.

$$\begin{bmatrix} \mathrm{Re}[K-\mu M] & -\mathrm{Im}[K-\mu M] & \mathrm{Re}\,[-Mu] & -\mathrm{Im}[-Mu] \\ \\ \mathrm{Im}[K-\mu M] & \mathrm{Re}[K-\mu M] & \mathrm{Im}\,[-Mu] & \mathrm{Re}\,[-Mu] \\ \\ \mathrm{Re}\,[u^* M] & -\mathrm{Im}\,[u^* M] & 0 & 0 \\ \\ 0\cdots 0 & 1\,0\cdots 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} \mathrm{Re}[\frac{\partial u}{\partial \theta}] \\ \\ \mathrm{Im}[\frac{\partial u}{\partial \theta}] \\ \\ \mathrm{Re}[\frac{\partial \mu}{\partial \theta}] \\ \\ \mathrm{Im}[\frac{\partial \mu}{\partial \theta}] \end{Bmatrix} = \begin{Bmatrix} \mathrm{Re}[-\left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u] \\ \\ \mathrm{Im}[-\left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u] \\ \\ \mathrm{Re}[\frac{1}{2} u^* \frac{\partial M}{\partial \theta} u] \\ \\ 0 \end{Bmatrix} \tag{4.41}$$

Now, let's say we want to optimize the group velocity of a wave. The group velocity is the gradient of the frequency, ω, with respect to the wavevector, γ. As with θ, it is helpful to think of γ as a scalar initially, then generalize to vector γ by repeating the computation for each component of the wavevector. First, we know from Eq. 4.29 that the gradient of the eigenvalue, ω^2, with respect to the wavevector, γ, for a given branch and wavevector can be determined by

$$\frac{\partial \mu}{\partial \gamma} = u^* (\frac{\partial K}{\partial \gamma} - \mu \frac{\partial M}{\partial \gamma}) u \tag{4.42}$$

Note that Eq. 4.42 gives $\frac{\partial \omega^2}{\partial \gamma}$, and does not directly give the group velocity, which is $\frac{\partial \omega}{\partial \gamma}$. To obtain the group velocity, we can use the chain rule. Consider two functions $f(\gamma)$ and $g(\gamma)$, such that $f(\gamma) = g(\gamma)^2$.

$$g(\gamma) = \sqrt{f(\gamma)}$$

$$\frac{\partial g(\gamma)}{\partial \gamma} = \frac{1}{2} f^{-\frac{1}{2}} \frac{\partial f}{\partial \gamma} \tag{4.43}$$

Letting $f(\gamma) = \mu = \omega^2$ (so $g(\gamma) = \omega$), we arrive at

$$\frac{\partial \omega}{\partial \gamma} = \frac{1}{2\omega} \frac{\partial \mu}{\partial \gamma} \tag{4.44}$$

Using Eq. 4.42, 4.44, we can write an analytical equation to compute the group velocity.

$$\frac{\partial \omega}{\partial \gamma} = \frac{1}{2\omega} u^* (\frac{\partial K}{\partial \gamma} - \mu \frac{\partial M}{\partial \gamma}) u \tag{4.45}$$

Note that this same equation can be used to compute the design sensitivity of the eigenfrequency, ω, (in contrast to Eq. 4.29, which computes the gradient of the eigenvalue, μ).

$$\frac{\partial \omega}{\partial \theta} = \frac{1}{2\omega} u^* (\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}) u \tag{4.46}$$

By taking the gradient of Eq. 4.42 with respect to a design vector, we can aim to obtain the design sensitivity that would be used for a gradient-based optimization of group velocity.

First, to simplify notation, define the matrix A as follows.

$$A = \frac{\partial K}{\partial \gamma} - \mu \frac{\partial M}{\partial \gamma} \tag{4.47}$$

Substituting into Eq. 4.45 shows

$$\frac{\partial \omega}{\partial \gamma} = \frac{1}{2\omega} u^* A u \tag{4.48}$$

Next, take the gradient with respect to a design vector, θ.

$$\begin{aligned} \frac{\partial}{\partial \theta}\left[\frac{\partial \omega}{\partial \gamma}\right] = \frac{\partial^2 \omega}{\partial \theta \partial \gamma} =& \frac{\partial}{\partial \theta}\left[\frac{1}{2\omega}\right] u^* A u \\ &+ \frac{1}{2\omega}\frac{\partial u^*}{\partial \theta} A u \\ &+ \frac{1}{2\omega} u^* \frac{\partial A}{\partial \theta} u \\ &+ \frac{1}{2\omega} u^* A \frac{\partial u}{\partial \theta} \end{aligned} \tag{4.49}$$

Next, let's note how to find some of the quantities needed to use Eq. 4.49.

- $\frac{\partial}{\partial \theta}\left[\frac{1}{2\omega}\right] = -\frac{1}{2\omega^2}\frac{\partial \omega}{\partial \theta} = -\frac{1}{2\omega^2}\frac{1}{2\omega} u^* \left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u = -\frac{1}{4\omega^3} u^* \left(\frac{\partial K}{\partial \theta} - \mu \frac{\partial M}{\partial \theta}\right) u$
- $\frac{\partial u}{\partial \theta}$ can be found by solving the linear system given by Eq. 4.41.
- $\frac{\partial A}{\partial \theta} = \frac{\partial}{\partial \theta}\left[\frac{\partial K}{\partial \gamma} - \mu \frac{\partial M}{\partial \gamma}\right] = \frac{\partial^2 K}{\partial \theta \partial \gamma} - \frac{\partial \mu}{\partial \theta}\frac{\partial M}{\partial \gamma} - \mu \frac{\partial^2 M}{\partial \theta \partial \gamma}$

As for computing the derivatives of the system matrices K, M, we can recall that they are constructed by applying wavevector-dependent transformation matrices to design-dependent full system matrices K_f, M_f.

$$\begin{aligned} K(\theta,\gamma) &= T^{*}(\gamma)K_f(\theta)T(\gamma) \\ M(\theta,\gamma) &= T^{*}(\gamma)M_f(\theta)T(\gamma) \end{aligned} \tag{4.50}$$

$$\begin{aligned} K &= T^{*}K_fT \\ M &= T^{*}M_fT \end{aligned} \tag{4.51}$$

$$\begin{aligned} \frac{\partial K}{\partial\gamma} &= \frac{\partial T^{*}}{\partial\gamma}K_fT + T^{*}K_f\frac{\partial T}{\partial\gamma} \\ \frac{\partial M}{\partial\gamma} &= \frac{\partial T^{*}}{\partial\gamma}M_fT + T^{*}M_f\frac{\partial T}{\partial\gamma} \end{aligned} \tag{4.52}$$

$$\begin{aligned} \frac{\partial K}{\partial\theta} &= T^{*}\frac{\partial K_f}{\partial\theta}T \\ \frac{\partial M}{\partial\theta} &= T^{*}\frac{\partial M_f}{\partial\theta}T \end{aligned} \tag{4.53}$$

$$\begin{aligned} \frac{\partial^2 K}{\partial\theta\partial\gamma} &= \frac{\partial T^{*}}{\partial\gamma}\frac{\partial K_f}{\partial\theta}T + T^{*}\frac{\partial K_f}{\partial\theta}\frac{\partial T}{\partial\gamma} \\ \frac{\partial^2 M}{\partial\theta\partial\gamma} &= \frac{\partial T^{*}}{\partial\gamma}\frac{\partial M_f}{\partial\theta}T + T^{*}\frac{\partial M_f}{\partial\theta}\frac{\partial T}{\partial\gamma} \end{aligned} \tag{4.54}$$

C h a p t e r 5

VISUAL VIBRATION TOMOGRAPHY: ESTIMATING INTERIOR MATERIAL PROPERTIES FROM MONOCULAR VIDEO

[1] Berthy T. Feng et al. “Visual Vibration Tomography: Estimating Interior Material Properties from Monocular Video”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 2022, pp. 16210–16219. DOI: `10.1109/CVPR52688.2022.01575`. URL: `https://doi.org/10.1109/CVPR52688.2022.01575` (visited on 04/19/2024).

Author contributions

A.C.O. contributed physical insights to the development of the method, developed numerical models in COMSOL for simulated experiments, contributed to the experimental validation, and participated in writing and revising the manuscript.

Chapter preface

This chapter marks a change of focus of this thesis, pivoting from the first main focus (*data-driven surrogate modeling of elastodynamic systems*) to the second main focus of study during my PhD: *inference problems for elastodynamic systems.* Wave propagation and vibration phenomena are used extensively to evaluate and characterize the interiors of objects. This is tremendously useful in medical situations, for example, to more precisely navigate delicate biological structures with ultrasound-guided inverventions. In seismology, these concepts are used to assess the mechanical state of different regions of soil to advise construction and earthquake engineering. In many applications of elastodynamic imaging techniques, sensors such as accelerometers or piezoelectric transducers are placed directly on the surface of the system being characterized in order to collect data for inference. However, there are many applications where it may not be feasible to outfit a system with a full set of sensors. Here, we explore a low-data approach to this problem, using only visual observations of the object under excitation to characterize the otherwise-unseen features of its interior.

This chapter introduces *Visual Vibration Tomography* (VVT), a method to mechanically characterize the full three-dimensional volume of an object by observing the vibration of the object through monocular video. The method extracts sub-pixel motions from the video to understand partial information about the the vibrational modes of the object, then uses optimization to converge on a physics-based model that best explains the modal observations. VVT is tested extensively in simulated experiments to understand the accuracy and limits of the approach, and demonstrates a real-world proof-of-concept in the characterization of two-dimensional inhomogeneous membrane systems and three-dimensional inhomogeneous solid systems.

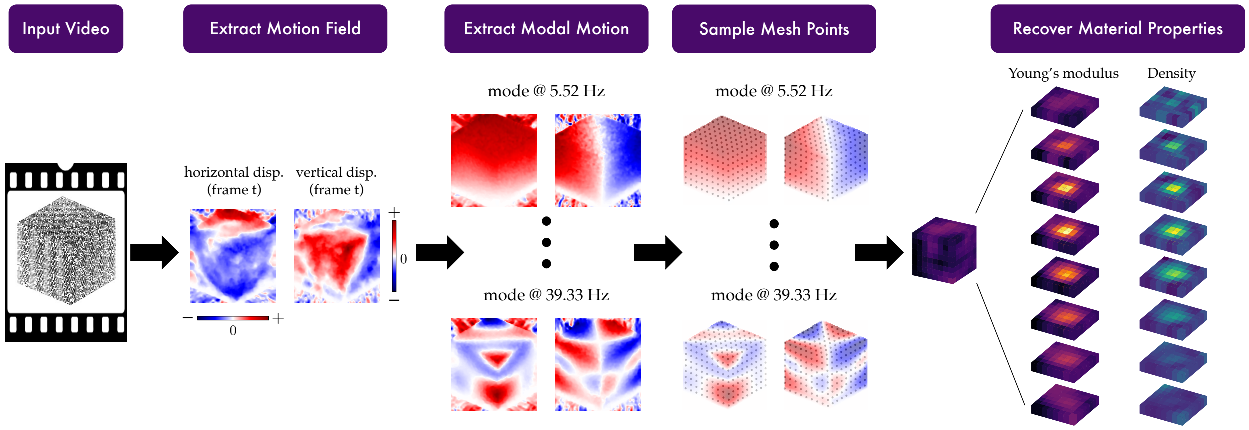


Figure 5.1: Method overview. Starting with a video showing vibration of an object, we extract the motion fields across time and then decompose this motion into image-space modes. From the image-space modes sampled at visible mesh points, we are able to recover a voxelized volume of the Young's modulus and density throughout the interior of the object.

5.1 Abstract

An object's interior material properties, while invisible to the human eye, determine motion observed on its surface. We propose an approach that estimates heterogeneous material properties of an object from a monocular video of its surface vibrations. Specifically, we show how to estimate Young's modulus and density throughout a 3D object with known geometry. Knowledge of how these values change across the object is useful for simulating its motion and characterizing any defects. Traditional non-destructive testing approaches, which often require expensive instruments, generally estimate only homogenized material properties or simply identify the presence of defects. In contrast, our approach leverages monocular video to (1) identify image-space modes from an object's sub-pixel motion, and (2) directly infer spatially-varying Young's modulus and density values from the observed modes. We demonstrate our approach on both simulated and real videos.

5.2 Introduction

The subtle motions of objects around us are clues to their physical properties. Among such properties are stiffness and density, which dictate how an object will respond to environmental forces. As humans, we can vaguely characterize how stiff or heavy a material is, such as when we infer that a rubber basketball will bounce higher than a ceramic bowling ball by tapping on its surface. Most engineering applications, however, require a greater level of detail, such as when an aeronautical engineer must faithfully simulate how an airplane wing will react to wind turbulence. In computer vision and graphics, a full characterization of an object's material properties allows one to faithfully simulate its behavior. These scenarios require non-destructive testing to obtain physical properties of the object without altering it.

We propose *visual vibration tomography*, a method to estimate material properties of an object directly from vibration signals extracted from monocular video. Much of non-destructive testing (NDT) has focused on measuring vibrations to identify the presence of defects in structures with a known geometry. However, NDT tools are not generally used to determine the precise spatial distribution of physical properties in objects with a heterogeneous interior structure.

We show that we can measure vibrations as sub-pixel motion in 2D video and then use this motion to constrain 3D material-property estimation. Videos have several advantages over existing NDT techniques: while contact sensors and laser vibrometers take point measurements, videos offer spatially dense measurements of surface vibrations. While laser vibrometers are expensive and specialized, cameras are ubiquitous and general-purpose. While existing image-based techniques require stereo cameras for 3D motion tracking, our method shows that in many cases, a monocular view is all you need.

Our motivating insight is that, under fixed geometry, an object's material properties determine its motion. The inverse direction is also true: motion determines material properties up to a scaling factor. If the motion is small, it can be decomposed into independent modes at natural frequencies, lending itself to a concise mathematical equation linking modes and material properties. This link lays the foundation for our physics-constrained optimization approach. The key challenge of our task is to deal with incomplete and 2D (as opposed to full-field) modes. Despite these challenges, we show that we can estimate material properties from image-space motion and recover full-field modes.

In this paper, we first review related work and the theoretical relationship between modes and material properties. We then show how to extract image-space modes from video and recover material properties (Fig. 5.1 shows an overview of the method). We demonstrate our approach on simulated data of 3D geometries and discuss the effects of damping and model mismatch. Finally, we present proof-of-concept experiments on real data, showing that we are able to image the shape of unseen material inhomogeneities on drum heads and the presence of a defect in a real 3D Jello cube. These experiments demonstrate promise for the future of the approach in more challenging environments. [1]

---

[1] Project website: `http://imaging.cms.caltech.edu/vvt`

5.3 Related Work

5.3.1 Material Analysis from Images and Video

In computer vision, scene understanding is an important goal that includes, among many tasks, characterization of materials. Previous work has estimated material categories [157–159] and surface properties [160, 161] from images. In contrast to static images, videos have been used to estimate material properties, although these are often restricted to specific object categories, such as fabrics [162–166] and trees [167]. Other work has inferred material properties from 3D point clouds [168, 169] and known external forces [170], but such measurements are harder to obtain than a 2D video. "Visual vibrometry" [171, 172] uses a video's motion spectrum to estimate stiffness and damping of fabrics and rods. This is a promising step towards a general approach for estimating material properties, but it is restricted to homogenized properties. In a similar vein, others have used video data to identify structural modes [173–175]. Davis et al. demonstrated how to visualize image-space modes and use them for plausible simulation [176, 177].

5.3.2 Traditional NDT

Non-destructive testing (NDT) is an umbrella term for any technique that collects data of a material or structure without damaging it. Usually, the goal is to identify defects or material inconsistencies that would change the expected behavior of the object. Laser vibrometry [178] and digital image correlation (DIC) [179] are popular tools for measuring surface displacements. Laser vibrometry has been used to examine the integrity of building structures [180, 181] and materials [182, 183]. DIC also has been used to identify defects in 2D structures [184–188]. Both laser vibrometry [189, 190] and DIC [191, 192] can be used for modal analysis, which involves identifying modal frequencies and shapes of a structure. While usually regarded as a verification tool rather than a means to directly infer material properties, recovered modal information has been used to solve for homogenized material properties [171, 193]. However, to our knowledge, modal analysis has not been used to solve the more challenging inverse problem of quantifying the heterogeneous properties addressed in this paper.

5.4 Background

5.4.1 Modal Analysis

Every object has resonant, or natural, frequencies. At each *resonant frequency*, the object vibrates periodically in a particular shape, called a *mode*. The vibration of a linear elastic object can be decomposed into independent modes.

In the finite element method (FEM), we model an object as a mesh, composed of elements that each take on material-property values. The mechanical properties that determine an object's vibration are Young's modulus (E), Poisson's ratio (ν), and density (ρ). E and ν define the stiffness of connections between vertices, while ρ defines the mass distribution. In this discretized model, the $n \times n$ stiffness matrix K describes the stiffness between each pair of n total DOFs, and the $n \times n$ mass matrix M describes the mass concentrated between each pair of DOFs. A mode u and frequency ω are an eigenvector-eigenvalue solution of the generalized eigenvalue problem:

$$Ku = \omega^2 Mu. \tag{5.1}$$

As Fig. 5.2 illustrates, a small change in material properties (within a fixed geometry) results in a small change in modal motion. As most solid materials have Poisson's ratio ≈ 0.3 [194], the principal properties affecting motion are Young's modulus, which determines K, and density, which determines M. *Our method is based on the insight that mode shapes on the surface of an object may reveal internal spatial inhomogeneities in these properties.*

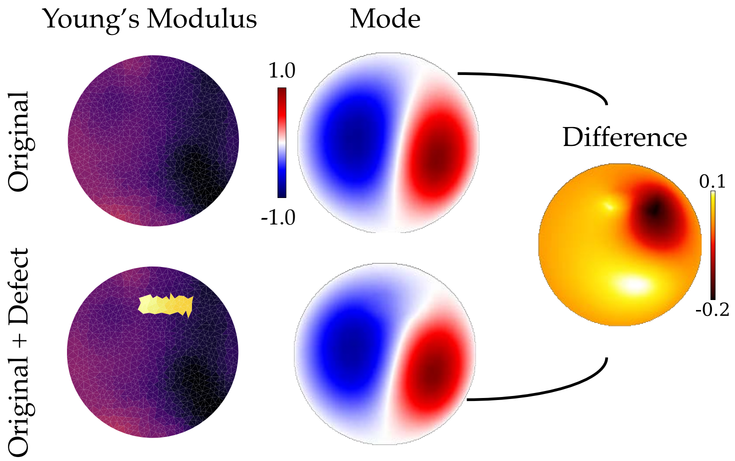


Figure 5.2: Small changes in material properties affect motion. Here a small region of a circular membrane becomes stiffer from "Original" to "Original + Defect." This change appears as a slight change in the mode shown. We propose using small changes in observed modal motion to recover the locations and shapes of defects.

5.4.2 Challenge of Monocular Material Estimation

We begin by setting up a simplified version of the inverse problem. Assuming we perfectly measure all modes u and frequencies ω, then by Eq. 5.1, we have the following minimization problem:

$$K^*, M^* = \arg\min_{K,M} \|KU - MU\Lambda\|_2^2, \tag{5.2}$$

where U is the matrix whose columns are modes u, and Λ is the diagonal matrix containing eigenvalues ω^2. For a known geometry, this is a convex problem with respect to K and M. However, Eq. 5.2 requires that we have access to all 3D modes and frequencies. In contrast, we will be working with experimentally-observed, image-space modes, incurring the following challenges:

1. *Unseen degrees of freedom (DOFs).* We typically only observe a fraction of an object. For

example, when observing a 3D cube with a monocular camera, one can see at most three of its sides, projected onto two directions of motion. Consider an 8x8x8 cubic mesh, which has $(8+1)^3 = 729$ vertices. With three directions of motion, it has $3 \times 729 = 2187$ total DOFs. But a single monocular view of three sides of the cube can only observe 217 vertices, moving in two directions of motion, amounting to $2 \times 217 = 434$ image-space DOFs. This alone limits us to observing *fewer than 20%* of the full-field DOFs for an 8x8x8 cube.

2. *Unseen modes.* Theoretically, for discrete meshes, there are as many modes as there are DOFs. However, we can only capture modes at frequencies below the Nyquist sampling rate of the camera, which is FPS/2.
3. *Noise.* Aside from camera noise, there is noise from motion extraction, particularly in non-textured regions.

Due to limited data, the problem of solving for K and M (Eq. 5.2) is ill-posed.[2] As Fig. 5.3 shows, observed data typically accounts for a tiny fraction of the matrices involved.

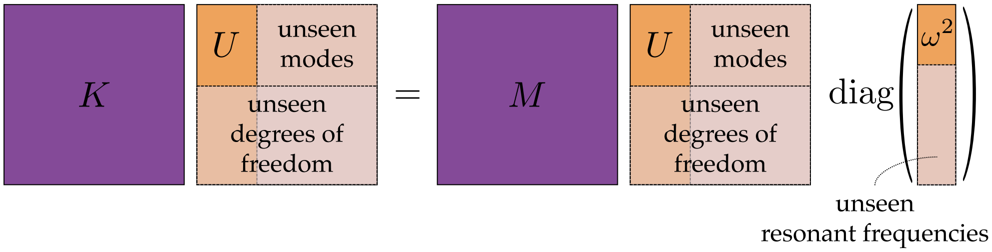


Figure 5.3: The generalized eigenvalue equation (Eq. 5.1) defines the relationship between K, M and U, $\boldsymbol{\omega^2}$. The matrix U has columns corresponding to modes and rows corresponding to DOFs. The vector $\boldsymbol{\omega^2}$ contains associated eigenvalues. We would like to solve for K and M given partial information about U and ω^2.

5.5 Approach

Our aim is to use motion features from a video to estimate material properties. This involves two stages: (1) motion extraction and image-space mode identification, and (2) solving for material properties that best match the observed image-space modes. The input is a video of an object, of which a mesh is known, and we assume that it is vibrating under linear elasticity (i.e., small motion). The output is two 3D images showing voxelized Young's modulus and density values throughout the object.

---

[2] For a known geometry and complete mode and eigenvalue information, K and M are fully determined up to a scaling factor.

5.5.1 Extracting Image-Space Modes from Video

Motion extraction: Since our approach relies on small, often imperceptible, motions, we need a way to extract sub-pixel motions from video. To quantify the displacements, we use the phase-based approach of Wadwha et al. [195], which computes local phase shifts in a complex steerable pyramid [196–198]. This method has the advantage over other tracking methods (e.g., optical flow) of being robust to tiny motion, down to 0.001 pixel [171]. The phase shifts are converted to pixel displacements using the approach proposed by Fleet and Jepson [199, 200]. To increase the signal-to-noise ratio, we filter out outlier pixels (i.e., top 1% of displacement magnitudes) and then apply an amplitude-weighted Gaussian blur. The result of this step is a motion field for each frame, which quantifies the horizontal and vertical displacement at each pixel relative to the first frame.

Identifying image-space modes: Modes are simply periodic motions occurring at particular frequencies, so we would expect them to appear as peaks in the power spectrum of motion amplitude. As is done in previous work that extracts image-space modes [172, 176, 177], we perform a discrete Fourier transform on the motion fields to analyze them in frequency space. To make this more concrete, let $\Delta x_t(x, y)$ and $\Delta y_t(x, y)$ be the horizontal and vertical displacements, respectively, at each pixel (x, y) at frame t. The 1D FFT across time of these displacement fields results in complex-valued $\widehat{\Delta x}_\ell(x, y)$ and $\widehat{\Delta y}_\ell(x, y)$, corresponding to frequencies $f_\ell = (\text{FPS} \cdot \ell/T)$ Hz for $\ell \in [1, T]$. The motion power at frequency f_ℓ is then defined as $\left\| \left[\widehat{\Delta x}_\ell, \widehat{\Delta y}_\ell \right] \right\|_2^2$ (dropping (x, y) notation for clarity). A peak ℓ^* in the power spectrum ideally corresponds to a natural frequency f_{ℓ^*} and image-space mode given by $\left[\text{Re}\left(\widehat{\Delta x}_{\ell^*} \right), \text{Re}\left(\widehat{\Delta y}_{\ell^*} \right) \right]$.

Sampling image-space modes at mesh vertices: To approximate the 3D-to-2D projection matrix, a user manually identifies the pixel locations of several "reference" mesh vertices, and P is the projection matrix that best maps the corresponding mesh coordinates to the image. Using P, we map all of the mesh vertices from their 3D coordinates to 2D image coordinates. We then sample each image-space mode at the pixel locations of *visible* mesh vertices. For mode j, we construct a vector γ_j that contains the horizontal and vertical displacements of each mesh vertex at the corresponding natural frequency. Supposing we observe q' out of q mesh vertices, the vector γ_j has the form

$$\gamma_j = \left[\Delta x_1, \Delta y_1, \ldots, \Delta x_{q'}, \Delta y_{q'}, 0, \ldots, 0 \right]^\intercal \in \mathbb{R}^{2q}, \tag{5.3}$$

where Δx_i is the horizontal (pixel) displacement and Δy_i the vertical (pixel) displacement of vertex i. Unseen vertices are assigned displacements of 0, and for notational clarity, we position them at the end of the vector.

5.5.2 Estimating Material Properties

The matrices K and M are functions of Young's modulus and density. While typically expressed as *global* matrices, they can be decomposed into *local* matrices, which scale linearly with local material properties. As a result, K and M can each be written as a weighted sum of "unit" local matrices. Specifically, we voxelize the volume containing the mesh so that each voxel contains a sub-collection of mesh elements. Given Young's modulus w_e and density v_e for each voxel, we express the global matrices as

$$K = \sum_{e=1}^{m} w_e K_e \quad \text{and} \quad M = \sum_{e=1}^{m} v_e M_e, \tag{5.4}$$

where K_e and M_e are "unit" local stiffness and mass matrices, which we assemble using `FEniCS` [201], and m is the number of voxels. This allows us to represent K and M as functions of vectors $w, v \in \mathbb{R}^m$.

5.5.2.1 Optimization Formulation

Data-matching objective: Suppose we have k modal observations, where $\widehat{\gamma}_i$ and $\widehat{\omega}_i$ are the i-th observed image-space mode and natural frequency, respectively. We would like to determine the voxel-wise Young's modulus values w and density values v that, when assembled into global stiffness and mass matrices, result in 3D modes $u_1, \dots, u_k$ that agree with $\widehat{\gamma}_1, \dots, \widehat{\gamma}_k$ when projected onto image-space. Since we do not know the full-field 3D modes, we need to include them as decision variables. Intuitively, the data-matching objective is to minimize $\|Pu_i - \widehat{\gamma}_i\|$ for each i.

Regularization: To make the solution well-defined, we choose to minimize total squared variation (TSV) of w and v, which encourages spatial smoothness. Moreover, since we are estimating both stiffness and mass, the objective function can become arbitrarily low if we do not constrain the range of material-property values; this is because scaling K and M by a factor of s still satisfies the generalized eigenvalue equation: $(sK)u = \omega^2(sM)u$. To resolve this ambiguity, we choose to minimize the deviation of w from a mean value $\bar{w}$. Regardless of $\bar{w}$, the relative differences in w^*, v^* will not change, and for defect characterization, we generally only care about relative changes.

The resulting optimization problem is written as

$$
\begin{aligned}
w^*, v^* = \underset{\substack{w, v \in \mathbb{R}^m \\ K, M \in \mathbb{R}^{n \times n} \\ u_i \in \mathbb{R}^n, i=1,\ldots,k}}{\arg\min} & \left\{ \frac{\alpha_u}{2k} \sum_{i=1}^{k} \|Pu_i - \widehat{\gamma}_i\|_2^2 \right. \\
& \left. + \frac{\alpha_w}{2m} \left\|\nabla^2 w\right\|_2^2 + \frac{\alpha_v}{2m} \left\|\nabla^2 v\right\|_2^2 + \left(\sum_{e=1}^{m} w_e/m - \bar{w} \right)^2 \right\} \\
\text{s.t. } & K = \sum_{e=1}^{m} w_e K_e, \quad M = \sum_{e=1}^{m} v_e M_e, \\
& Ku_i = \widehat{\omega}_i^2 M u_i, \; i = 1, \ldots, k,
\end{aligned}
\tag{5.5}
$$

where α_u, α_w, and α_v are hyperparameters that balance the objective terms. The effects of the regularization weights (α_w and α_v) are discussed in the supplementary material.

5.5.2.2 Optimization Strategy

As defined in Eq. 5.6, we approximately solve Eq. 5.5 via a dual formulation of the problem. The eigen-constraints in Eq. 5.5 are too strict to enforce directly, so we incorporate them as quadratic penalties in the dual problem. The weight of each penalty term is a dual variable, y_i, and we apply dual ascent to gradually increase these penalty weights.

$$
\begin{aligned}
w^*, v^* = \underset{\substack{w, v \in \mathbb{R}^m \\ K, M \in \mathbb{R}^{n \times n} \\ u_i \in \mathbb{R}^n, i=1,\ldots,k}}{\arg\min} & \left\{ \frac{1}{2k} \sum_{i=1}^{k} y_i \left\|Ku_i - \widehat{\omega}_i^2 M u_i\right\|_2^2 \right. \\
& + \frac{\alpha_u}{2k} \sum_{i=1}^{k} \|Pu_i - \widehat{\gamma}_i\|_2^2 \\
& \left. + \frac{\alpha_w}{2m} \left\|\nabla^2 w\right\|_2^2 + \frac{\alpha_v}{2m} \left\|\nabla^2 v\right\|_2^2 + \left(\sum_{e=1}^{m} w_e/m - \bar{w} \right)^2 \right\} \\
\text{s.t. } & K = \sum_{e=1}^{m} w_e K_e, \quad M = \sum_{e=1}^{m} v_e M_e.
\end{aligned}
\tag{5.6}
$$

Eq. 5.6 is a non-convex problem, but it is quadratic with respect to w, v for fixed u_i, and it is quadratic with respect to u_i for fixed w, v. Our procedure is to iteratively compute the closed-form solution for $U = [u_1 \ldots u_k]$ and then $z = [w^\top, v^\top]^\top$, thereby minimizing the objective function at each step. We update the dual variables according to

$$
y_i^{t+1} = y_i^t + \eta \left\|K^{t+1} u_i^{t+1} - \widehat{\omega}_i^2 M^{t+1} u_i^{t+1}\right\|_2, \tag{5.7}
$$

where $\eta > 0$ is the dual-variable update rate. Once the decision variables have converged, we output the minimizing solution $z^* = [w^{*\top}, v^{*\top}]^\top$. w^* and v^* are the voxel-wise estimated Young's modulus and density values.

5.6 Simulated Experiments

We test our approach on the simulated vibration of 3D cubes with “defects," and we discuss the practical concerns of complex geometries, model mismatch, and damping.

5.6.1 Creating Synthetic Data

Cube model: We model a cube as a 10x10x10 hexahedral mesh, similar to a voxel grid. Each of the 1000 voxels is assigned a Young’s modulus and density that correspond to either the primary material or a defect material. The material properties are chosen to resemble Jello and clay, respectively ($E_{\text{jello}} = 9000$ Pa, $\rho_{\text{jello}} = 1270$ kg/m3, $E_{\text{clay}} = 5 \times 10^6$ Pa, $\rho_{\text{clay}} = 7620$ kg/m3). We set a homogeneous Poisson’s ratio of $\nu = 0.3$ [194].

Vibration animation: Once the cube’s mesh and material properties have been defined, we run a transient analysis in COMSOL [202], a commercial FEM software. The analysis calculates the cube’s deformation over time given an initial condition. We choose an initial condition that mimics “plucking” a corner of the cube (e.g., an initial displacement vector of $(0.5, 0.5, 0.5)$ cm of the top-front corner) and keep the bottom surface fixed. The resulting simulation represents free vibration with a Dirichlet boundary condition. The simulation is 6 seconds long at 2000 FPS. From the calculated displacements, we create an animation of the cube deforming over time by plotting the motion of random points on the surface of the cube with `matplotlib` [203].

5.6.2 Implementation and Evaluation Details

Mode selection: We use `scipy`’s [204] peak-finder to automatically identify peaks in the log-power spectrum of motion amplitude, as described in Sec. 5.5.1. For a given simulation, this leads to around 20–30 selected peaks. Most peaks correspond to either a true mode or a linear combination of true modes whose frequencies fall in the same FFT frequency bin. However, a few peaks do not correspond to a true mode; we include these false modes in the synthetic results to best mimic analysis of real videos.

Inference cube mesh: We infer on an 8x8x8 hexahedral mesh. Since the simulations are done with a 10x10x10 mesh, our results indicate robustness to a slight mesh mismatch. In the results presented, the simulation model and inference model use linear elements.

Hyperparameters: For every presented result from simulated data, $\alpha_w = 10^{-10}$, $\alpha_v = 10^{-7}$, and $\bar{w} = 9000$. Keeping these hyperparameters fixed, we ran a hyperparameter search on a dataset of 12 cubes with various defects to identify good values for α_u (Eq. 5.6) and η (Eq. 5.7). After testing all combinations of $\alpha_u \in \{1, 10, 100, 1000\}$ and $\eta \in \{0.1, 0.5, 1, 2, 5, 10\}$, we determined to set $\alpha_u = 10$ if the number of input modes is ≥ 10; otherwise, $\alpha_u = 1$. The dual variables y are always initialized to 1, with $\eta = 1$. The decision variables w and v are initialized to homogeneous values

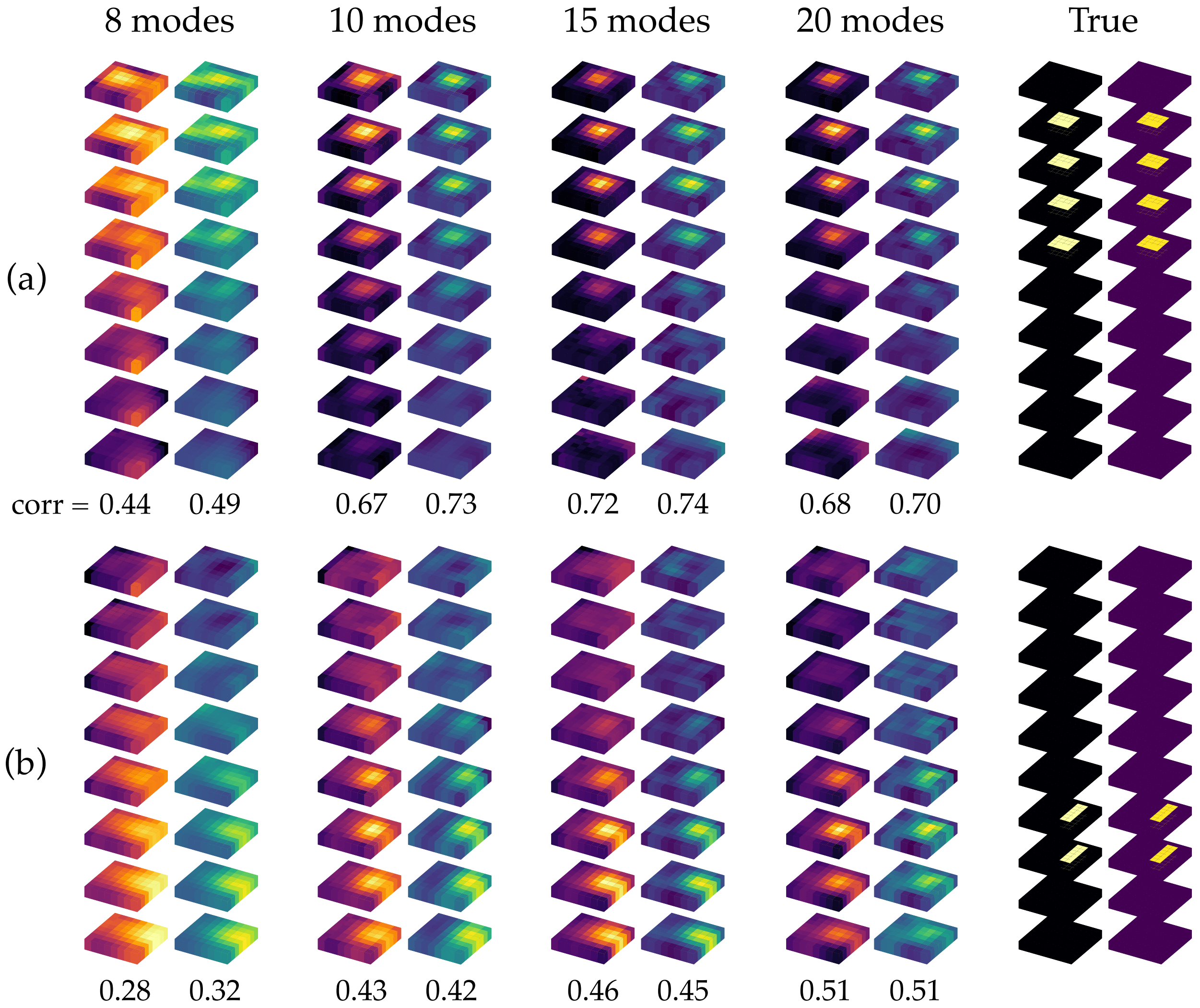


Figure 5.4: Reconstruction on two synthetic cubes with different defects. The given motion-extracted image-space modes range from the 8–20 lowest extracted modes. Normalized correlation generally increases as the number of modes increases. (b) is more challenging because the defect is smaller and closer to the bottom of the cube, where there is no motion.

of 9000 [Pa] and 1270 [kg/m3], respectively (the true values of the primary material).

Evaluation: Our method recovers *relative changes* in material properties (see Sec. 5.5.2.1). As such, the normalized 3D images of estimated Young's modulus and density should match the true normalized properties. We use *normalized correlation* between the estimated image and ground-truth image as the reconstruction score. Another way to assess estimated properties is to verify that they produce the same image-space modes and natural frequencies as the true properties (see supplementary material).

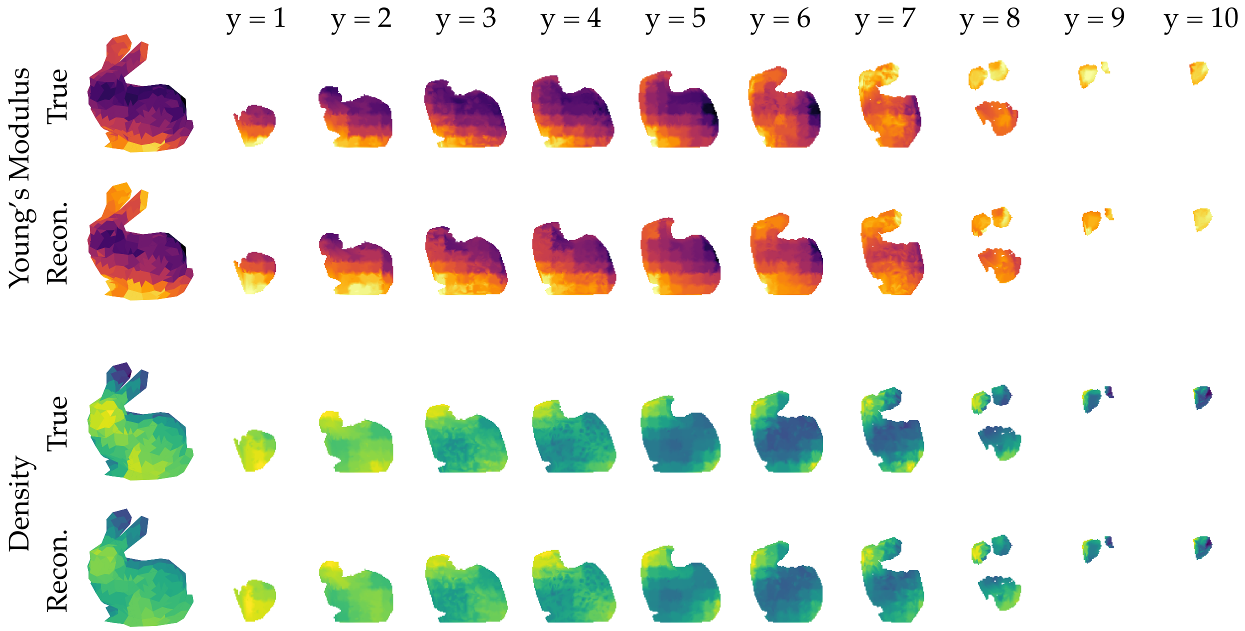


Figure 5.5: Reconstruction for the Stanford Bunny from (true) image-space modes. Slices along the y-axis are shown.

5.6.3 Results

Fig. 5.4 shows results for two different cubes with defects appearing at different locations. These results are obtained from noisy, motion-extracted image-space modes. As more modes are observed, the inverse problem becomes better constrained, sharpening the image of the interior defect. Also note that a defect near the top of the cube is easier to identify than one near the bottom. This is because the base of the cube is fixed and thus provides less motion signal.

Complex geometry: To demonstrate the approach on a more complex geometry, Fig. 5.5 shows a volumetric reconstruction of material properties for the Stanford Bunny [205]. We voxelize the volume containing the tetrahedral mesh[3] of the bunny into an 8x8x8 grid and match each mesh element to the nearest voxel, resulting in about 21 elements per voxel. 20 true image-space modes of the monocular view of the bunny shown are used for this reconstruction.

We next consider some challenges that may arise with real-world data: geometric mismatch and damping. More investigations into model mismatch are provided in the supplementary material.

5.6.3.1 Geometric Mismatch

Fig. 5.6 shows what happens when the dimensions of the inference mesh do not match the cube's true dimensions. We gradually increase the length of the inferred cube geometry in the x-direction from 1 to 1.4 times the true length. Scaling the length in one direction results in a gradually

---

[3] The bunny surface mesh is from `https://www.thingiverse.com/thing:151081`, and tetrahedralization is done with TetWild [206].

degrading estimate of the defect size. However, even with 30% geometric error, we are still able to distinguish that there is a defect located in the central region of the cube.

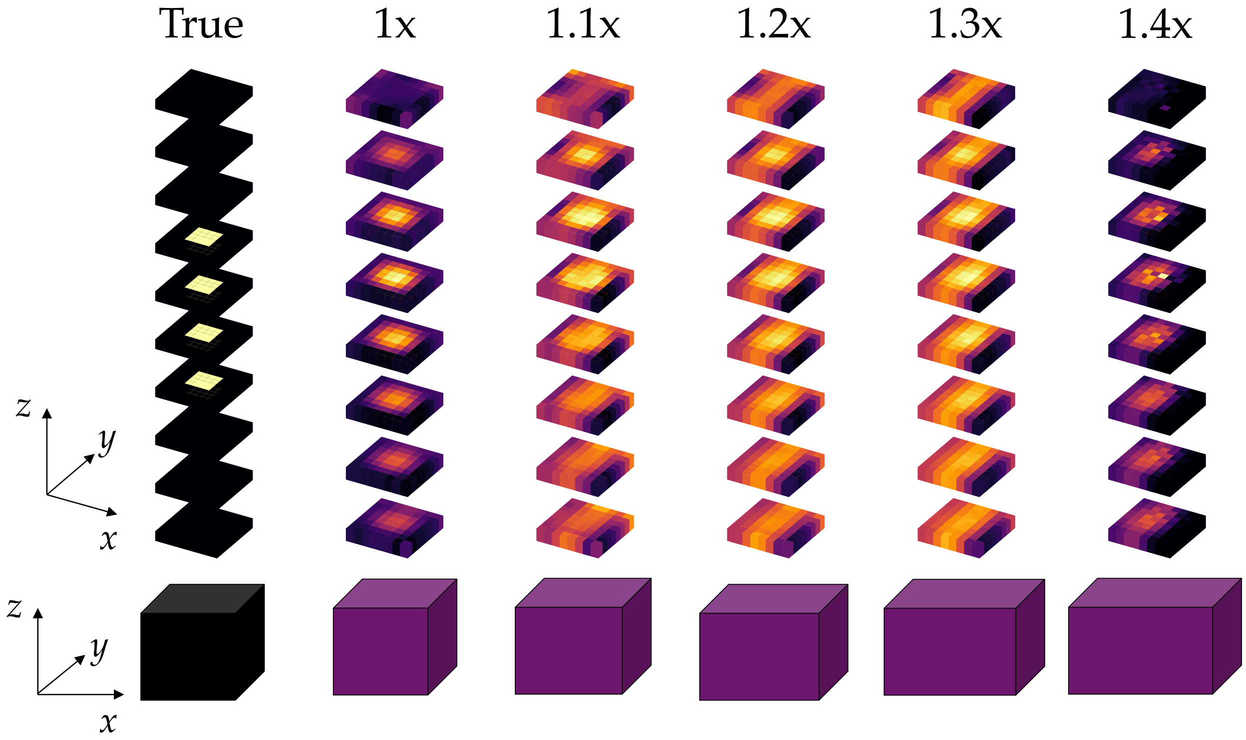


Figure 5.6: Geometric model mismatch. From 10 motion-extracted image-space modes, we infer on a mesh of various incorrect geometries, extending the inferred geometry width by a multiple of the true width. Results on Young's modulus show that some geometric mismatch can be accommodated.

5.6.3.2 Damping

Real-world objects exhibit various types of damping, which can affect both the frequencies and relative phases of its modes. To simulate damping, we incorporate Rayleigh damping into our synthetic cubes. Our damping parameters were estimated following the procedure outlined by Davis and Bouman et al. [171], who fit a Lorentzian curve to a peak in the motion power spectrum to estimate the damping ratio. We find that Jello cubes exhibit significant damping: from a real video of one, we estimated critical damping ratios of 0.01749 at 12.5 Hz and 0.01999 at 15.5 Hz.

Through realistic simulations in COMSOL, we find that damping poses the additional challenge of fewer observable modes. We can increase the number of observed modes by extracting modes from multiple simulations with different initial conditions. For example, “plucking" the top-back corner of a cube will cause slightly different modal expression than “plucking" its top-front corner.

Fig. 5.7 shows reconstruction for damped cubes. From two different plucking conditions, we are able to extract between 7–15 modes and use these modes to coarsely reconstruct the defect.

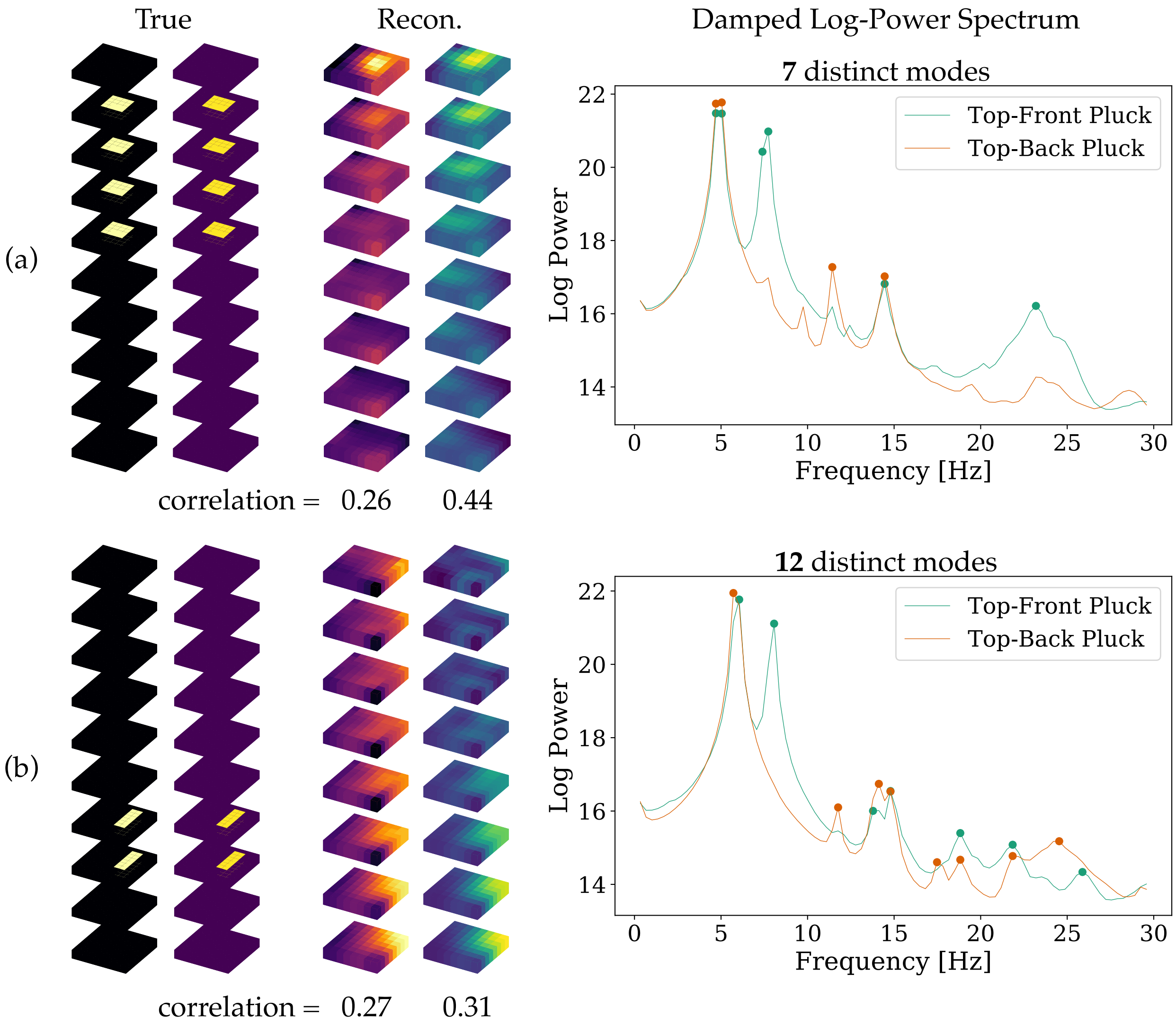


Figure 5.7: Reconstructions from two simulated damped cubes. Animations (3 seconds at 2000 FPS) of two different forcings are done: (1) a small initial displacement of the cube's top-front corner ("Top-Front Pluck") and (2) a small initial displacement of its top-back corner ("Top-Back Pluck"). Modes (marked as dots on the line plot) are selected based on the log-power spectrum of motion amplitude. Asymmetry plays a role in determining how many distinct modes are observable. As a cube becomes more asymmetric in its material-property distribution, its repeated eigenfrequencies become more separated. Since the defect in (b) is more off-center than the defect in (a), more distinct modes are identifiable in those simulations. In (a), with only 7 observed image-space modes, the reconstruction quality is consistent with Fig. 5.4, which shows only a coarse defect reconstruction when given 8 modes. (Note: in (b), although the number of observed modes is > 10, we show the reconstruction for $\alpha_u = 1$ instead of 10.)

5.7 Real-World Experiments

To demonstrate the potential of our approach in the real world, we applied it to real videos of drum heads and Jello cubes. With drum heads, we achieve reconstructions that allow one to discern distinct defect shapes, providing a proof-of-concept for defect discovery and characterization using our approach. The damping of 3D Jello cubes poses a challenge for extracting enough image-space modes for high-fidelity defect reconstruction; nonetheless, we are able to identify heterogeneity in the cube. Please refer to the supplementary material for details about the experiment setups, inference models, and hyperparameters.

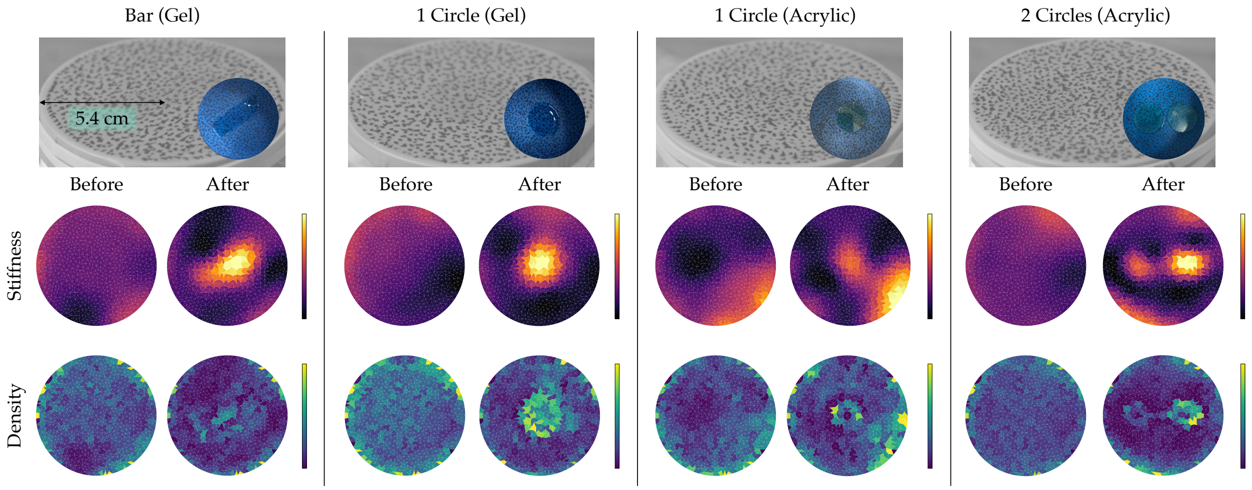

Bar (Gel)
1 Circle (Gel)
1 Circle (Acrylic)
2 Circles (Acrylic)

Figure 5.8: Reconstruction from real videos of drums. The defects shown are a gel bar, gel circle, acrylic circle, and two acrylic circles, applied to the underside of the drum head. For each defect, we recorded a video of the drum pre- and post-defect. One cannot see the defect in a video frame, but after applying our method, we were able to image the defects as changes in stiffness and density. For each type of defect, the “Before” and “After” material properties are plotted with the same normalized colormap.

5.7.1 Real Drums

We tested our method on a dataset of real drum heads, each altered with a defect beneath the surface. The defects were created from two materials: nail hardening gel (painted beneath the surface) or acrylic plastic circles (glued onto the bottom of the surface). Although all DOFs of the 2D membrane are visible in the video, solving for material properties is still ill-posed because we observe a limited number of projected modes (see Fig. 5.9).

Results: Fig. 5.8 shows estimated Young’s moduli and densities for various drum heads, before and after defects were included. For both materials, the defect appears as a bright region in stiffness. Interestingly, gel and acrylic appear differently in their density estimations. For gel defects, there is a bright, filled region in the density map that corresponds to a higher mass from the defect. For acrylic defects, this change only appears on the edges of the defect. This is possibly because

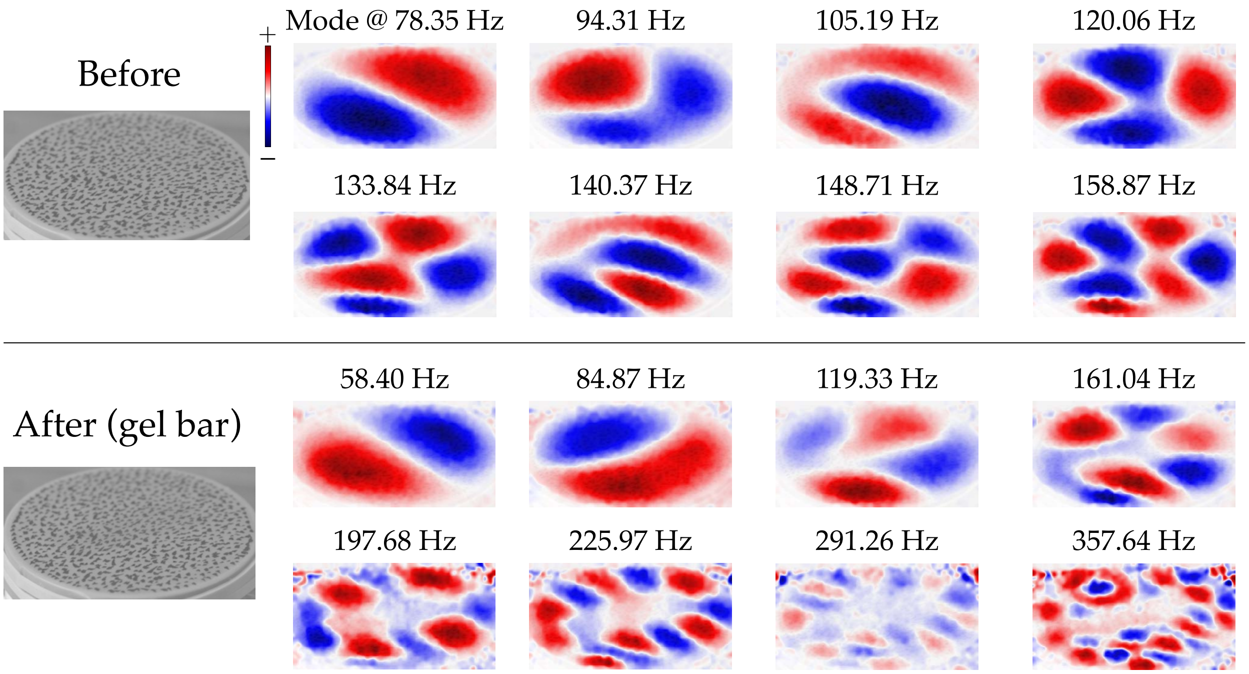


Figure 5.9: Extracted image-space modes from real videos of a drum, before and after a defect was introduced. The defect shown here is a gel rectangle, which was painted on the bottom of the drum head. Only vertical motion is shown.

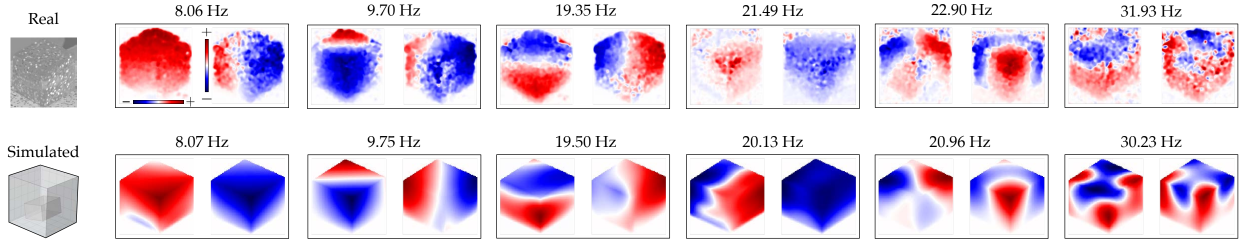


Figure 5.10: Extracted image-space modes from real videos of a Jello cube with an interior clay defect ("Real"). The true image-space modes identified from a COMSOL simulation of a cube with a defect are shown for comparison ("Simulated"). Each observed image-space mode has a corresponding simulated mode that appears similar in both image-space and eigenfrequency.

the acrylic circles are much stiffer than gel, which bends along with the rubber membrane. These results indicate that our proposed approach could be used to identify not just the presence of a defect, but also its shape.

5.7.2 Real Cubes

To gain further insight into practical challenges, we conducted an experiment on a real Jello cube with an interior clay defect. This object is more challenging than the drum membrane in two respects: (1) the high damping of Jello, perhaps due to its water content, and (2) the large proportion of unseen DOFs in the cube geometry. The cube had dimensions 4.9 x 4.7 x 4.5 cm,

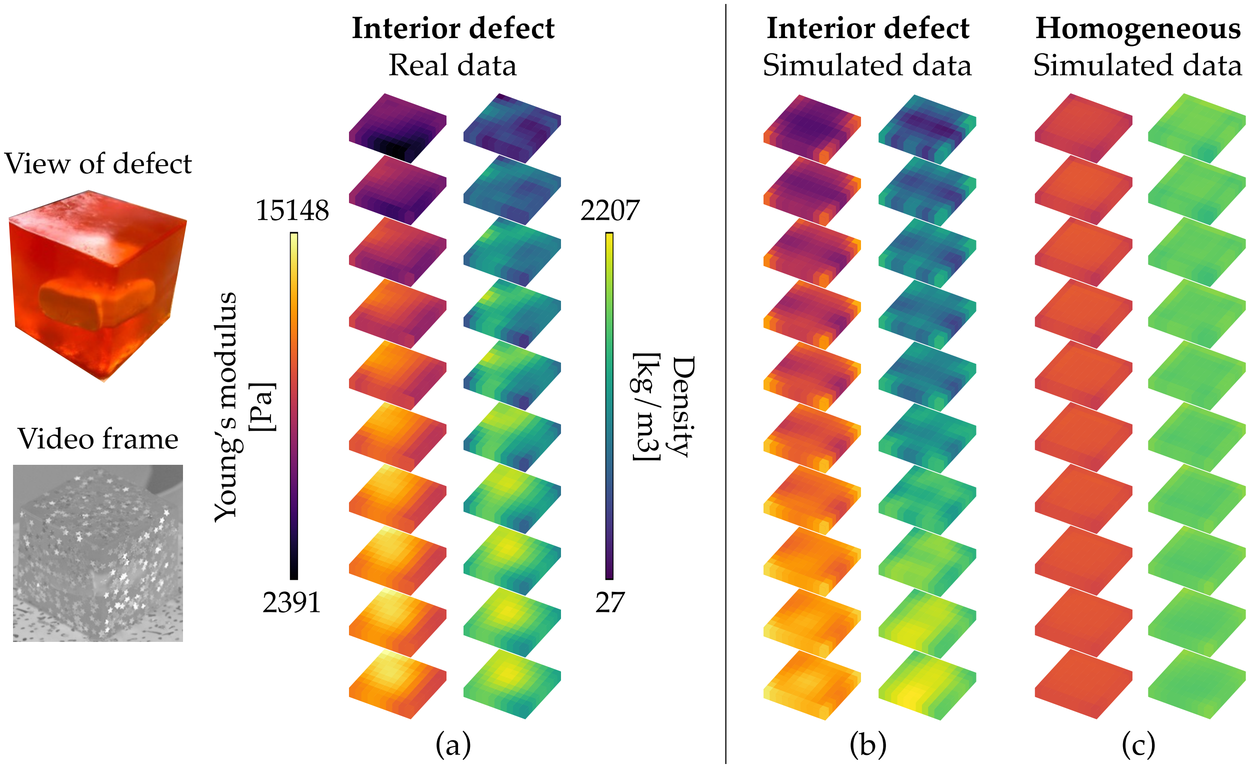


Figure 5.11: Reconstruction from real data vs. reconstructions from simulated data of the defect cube and a homogeneous cube. All reconstructions use 6 image-space modes and the same hyperparameters and are plotted with the same colormaps. As Fig. 5.10 shows, there is a one-to-one correspondence between the modes given for (a) and the modes given for (b). (a) is more similar to (b) than to (c), indicating that with 6 modes, we can differentiate between a cube with a defect and a homogeneous one.

while the rectangular clay defect was of size 2.2 x 2.9 x 1.4 cm. We recorded three videos of the cube under different initial deformation conditions (e.g., in one video, we lifted and then quickly released the top-front corner of the cube). Multiple videos allowed us to identify more unique modes and average duplicate ones.

We created two COMSOL models that would be comparable to the real Jello cube: one simulated cube had a clay defect and the other did not. The Young's modulus values of the Jello and clay were set so that the natural frequencies would agree with those observed. The Rayleigh damping parameters were estimated following the method mentioned in Sec. 5.6.3.2. As illustrated in Fig. 5.10, the COMSOL image-space modes of the simulated cube with a defect appear similar to those captured from the real Jello cube.

Results: Fig. 5.11 shows the result of our approach applied to real video data of the Jello cube. The reconstruction is obtained using six unique, motion-extracted image-space modes. As expected based on our findings in Figs. 5.4 and 5.7, we are able to recover only a large-scale estimation of material properties with six constraining modes. Still, it is very promising to have identified inhomogeneities in a real 3D object with our method. We further compare this real-data reconstruction to reconstructions obtained from simulated data of a homogeneous cube and one with a defect, showing that we achieve a solution that resembles the solution for the simulated defect cube more than it resembles the solution for the simulated homogeneous cube (Fig. 5.11).

We have shown one example of attaining simulation quality on a real cube. Further work needs to be done to achieve consistent results across a variety of objects. Additional camera views are one plausible, simple solution.

5.8 Limitations

Our method assumes that materials are isotropic and linear elastic. Linear elasticity is only satisfied if the object's motion is small. Further, we assume that the geometry is, at least roughly, known ahead of time (see Fig. 5.6).

For now, we have validated our method with a high-speed camera. We have not yet demonstrated the approach with consumer-grade cameras that bring additional challenges such as image compression and noise. Generally, the hardware required depends on the amplitude and frequencies of the modes. For large structures that vibrate below 100 Hz [207–209], a smartphone camera theoretically provides enough temporal frequency. Objects that vibrate more quickly require high-speed cameras. Tricks such as temporal aliasing via a strobe may expand the capabilities of a camera.

The primary challenge with applying this technique to real-world objects is capturing enough image-space modes to recover interior defects with high fidelity. Damping causes a reduction in the number of modes that can be extracted. We demonstrated that for damped Jello cubes, we could still recover some information from only six image-space modes (Sec 5.7.2). Even so, in many objects, damping will pose a more significant challenge. In the future, acquiring more modal observations could be solved by exciting modes through mechanical vibration tables.

5.9 Conclusion

We have shown that it is possible to recover spatially-varying material properties of 3D objects from monocular video, even in regions unseen in the image. This can be done by decomposing 2D surface motion into image-space modes, and then solving for the Young's modulus and density values that agree with the observed modes. We demonstrated our method on synthetic and real-world data of objects ranging from 2D drum heads to a 3D bunny.

Our results highlight that monocular videos are a simple, yet powerful, source of data for understanding the physical properties of objects around us. We believe that videos are a promising domain for further research into non-destructive testing, turning everyday visual sensors into tools for material characterization.

5.10 Supplementary

5.10.1 Optimization Runtime

Recall that the optimal material properties and 3D modes are found by iteratively updating the solutions to the following optimization problem:

$$
\begin{aligned}
w^*, v^* = \underset{\substack{w,v \in \mathbb{R}^m \\ K,M \in \mathbb{R}^{n\times n} \\ u_i \in \mathbb{R}^n, i=1,\dots,k}}{\arg\min} & \left\{ \frac{1}{2k} \sum_{i=1}^{k} y_i \left\| K u_i - \widehat{\omega}_i^2 M u_i \right\|_2^2 \right. \\
& + \frac{\alpha_u}{2k} \sum_{i=1}^{k} \| P u_i - \widehat{\gamma}_i \|_2^2 \\
& \left. + \frac{\alpha_w}{2m} \left\| \nabla^2 w \right\|_2^2 + \frac{\alpha_v}{2m} \left\| \nabla^2 v \right\|_2^2 + \left(\sum_{e=1}^{m} w_e/m - \bar{w} \right)^2 \right\} \\
\text{s.t. } K = \sum_{e=1}^{m} w_e K_e, \quad & M = \sum_{e=1}^{m} v_e M_e .
\end{aligned}
\tag{5.8}
$$

For an 8x8x8 cube with linear hexahedral elements, one iteration takes 2–3 seconds (tested on an 8-core Intel Core i9, 32 GB RAM), and convergence usually happens within 100 iterations.

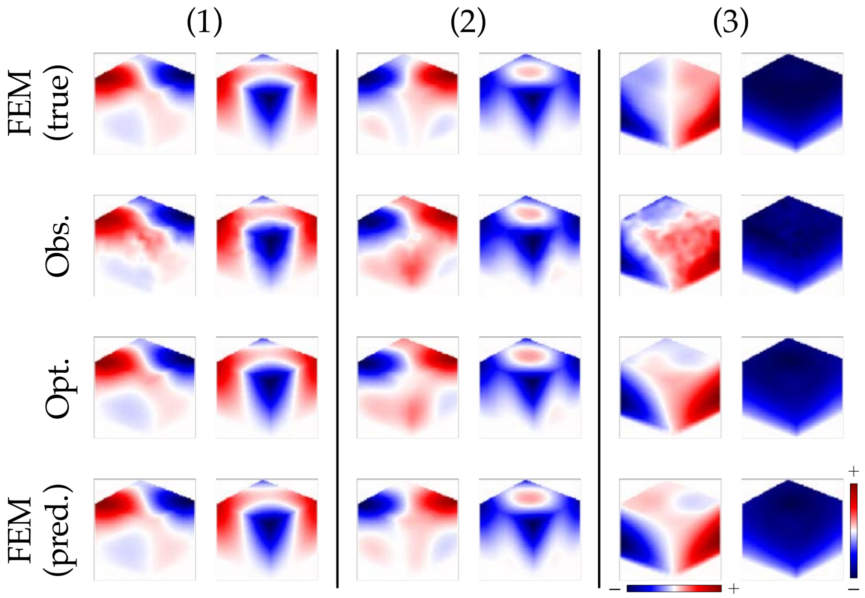
5.10.2 Simulated Experiments

5.10.2.1 Predicted Image-Space Modes

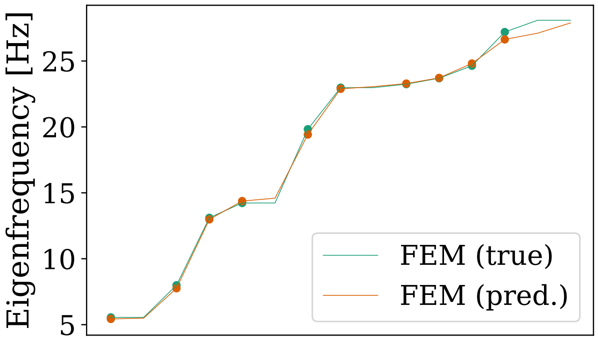
In addition to normalized correlation, a way to assess estimated material properties is to verify that they produce the same image-space modes and natural frequencies as the true properties. Recall that 3D modes are a decision variable in our optimization scheme (Eq. 5.8). As Fig. 5.12 shows, it is informative to compare the true FEM modes, observed modes, optimized modes, and predicted FEM modes, in image-space. The optimization process usually de-noises observed modes. For some modes, spatially correlated noise may make it difficult to recover the true mode, but it is possible for the predicted FEM modes to still be similar to the truth (see example (2) in Fig. 5.12).

5.10.2.2 Effect of Regularization

The strength of spatial regularization on material properties affects the smoothness of the estimation. In Eq. 5.8, the regularization weights for Young's modulus and density are α_w and α_v, respectively. In Fig. 5.13, we show how the estimated density image becomes sharper as α_v decreases (keeping all else fixed). This can result in a crisper picture of the defect, but can also make reconstruction more sensitive to noise.



(a)



(b)

Figure 5.12: (a) Similarity of predicted image-space modes to true, observed, and optimized image-space modes for the cube sample shown in Fig. 5.14. “Obs.” mode is often a noisy version of “FEM (true).” “Opt.” refers to the optimized solution U^* in Eq. 5.8. “FEM (pred.)” is the image-space mode resulting from the estimated material properties. (b) Predicted eigenfrequencies vs. true eigenfrequencies. The frequencies of the 10 given motion-extracted image-space modes are marked by scatter dots.

5.10.2.3 Intrinsic Resolution

Fig. 5.14 demonstrates how, as the number of input image-space modes increases, the intrinsic resolution of the reconstruction improves.

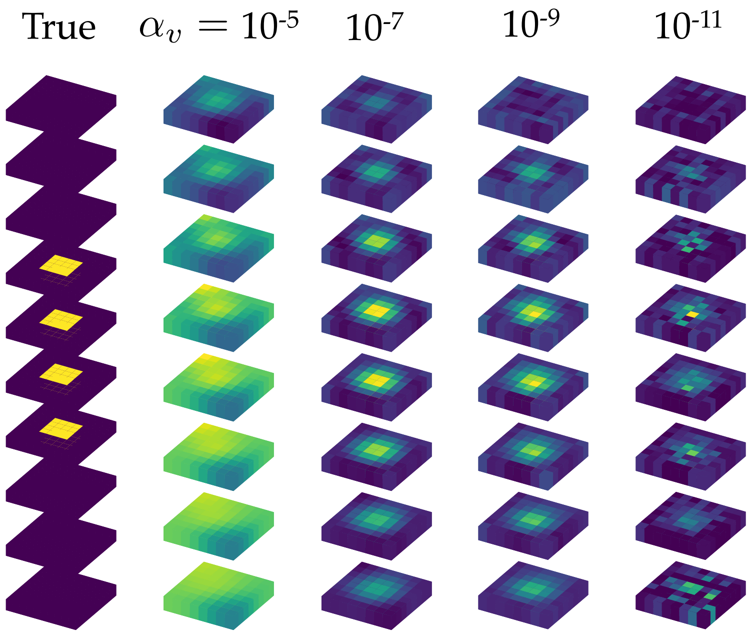


Figure 5.13: Effect of regularization. Here we show the estimated density for a cube sample (same as in Fig. 5.14). Keeping all else fixed, as α_v decreases, the image of the defect becomes crisper, but more sensitive to noise. Each estimation uses the same 10 motion-extracted image-space modes.

5.10.3 Simulated Experiments: Model Mismatch

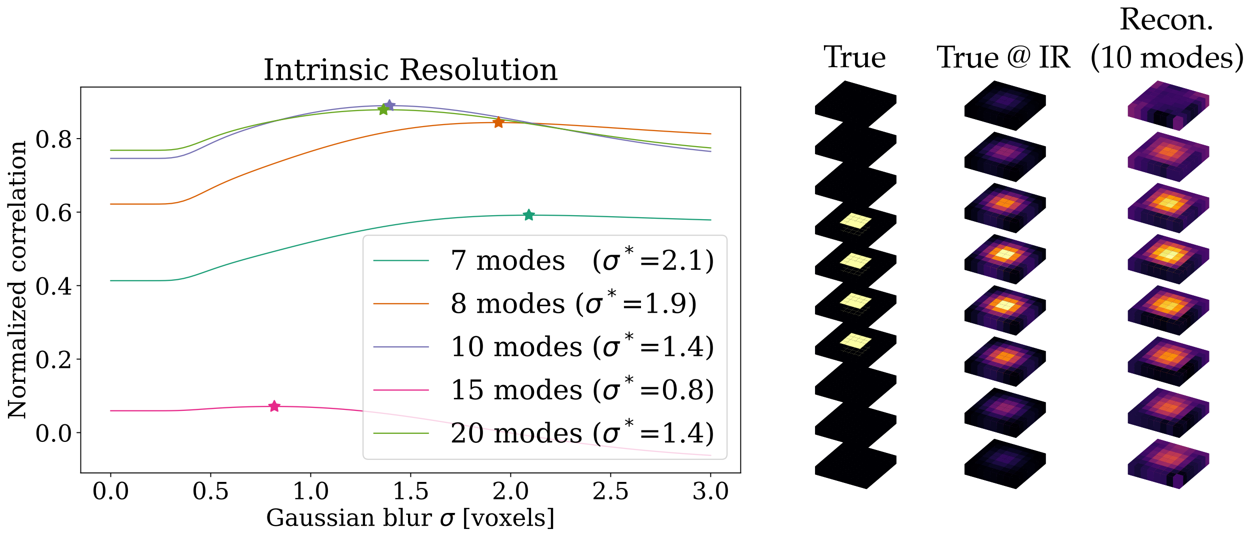
In the main material, we discussed geometry mismatch. Here, we investigate the effects of model mismatch in the Poisson's ratio and the mesh element order.

5.10.3.1 Poisson's Ratio Mismatch

The Poisson's ratio is a measure of the deformation of a material perpendicular to an applied force and ranges from 0.1 to 0.45 [210]. Since our optimization formulation only estimates Young's modulus and density, we assume that every voxel has the same Poisson's ratio. Fig. 5.15 shows that this assumption does not significantly hurt the reconstruction of a defect, especially when assuming a Poisson's ratio that is closer to that of the main material.

5.10.3.2 Mesh Element Order Mismatch

In general, it is better to use higher-order elements to model a real-life object. However, there is a tradeoff in efficiency. A quadratic element approximates node displacements more accurately, but contains more DOFs than a linear element. Fig. 5.16 shows what happens when the forward model uses quadratic elements, while the inference model uses linear elements. For real-world objects, one should choose the order that strikes the right balanace between approximation accuracy and computational cost.



(a)

(b)

Figure 5.14: Intrinsic resolution of reconstructed volumes. Based on normalized correlation with the ground-truth material properties smoothed at different scales, one can approximate the intrinsic resolution of the reconstructed material properties. In (a), we plot normalized correlation versus Gaussian blur standard deviation σ, for the reconstruction of Young's modulus using different numbers of image-space modes (keeping all other hyperparameters fixed). As the number of observed modes increases, the reconstructed resolution also increases (i.e., smaller σ). (b) shows the true Young's modulus image blurred at the intrinsic resolution (IR) of the reconstruction given 10 image-space modes ($\sigma^* = 1.4$ voxels).

5.10.4 Drum Experiment

5.10.4.1 Drum Construction

The drums were constructed by fixing a thin rubber sheet over a 4"x4" PVC adaptor with a rubber band. We tested defects of two materials: nail hardening gel and acrylic plastic circles. For each defect, we recorded a video of the homogeneous drum before the defect was applied for comparison. We drew a speckle pattern on the drum head for texture.

5.10.4.2 Vibration-Capture Setup

Fig. 5.17 shows a schematic of the setup. We taped the drum onto an optical table, with the high-speed camera standing on the same optical table. The excitation source was a PreSonus Sceptre S8 loudspeaker, which sat on a platform separate from the optical table and was pointed at the drum. For each video, we recorded the drum head's vibration in response to a 3.5-second linear frequency sweep (50–1000 Hz) played by the speaker.

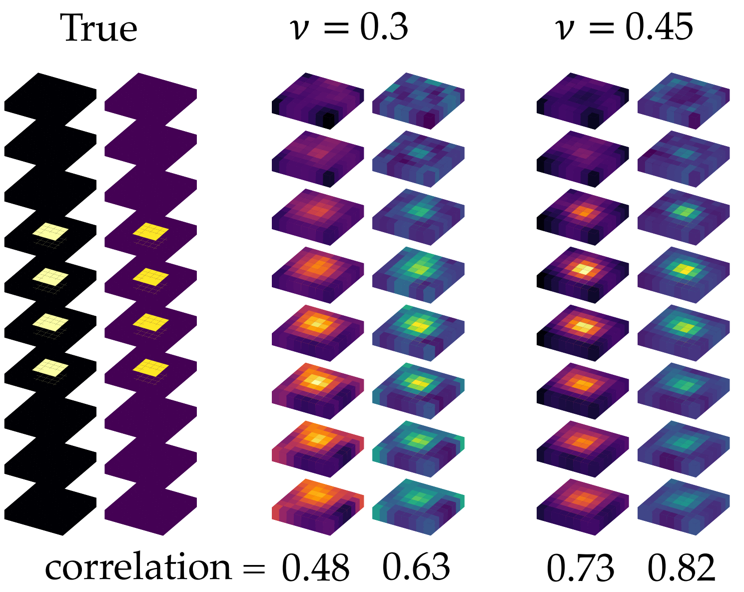


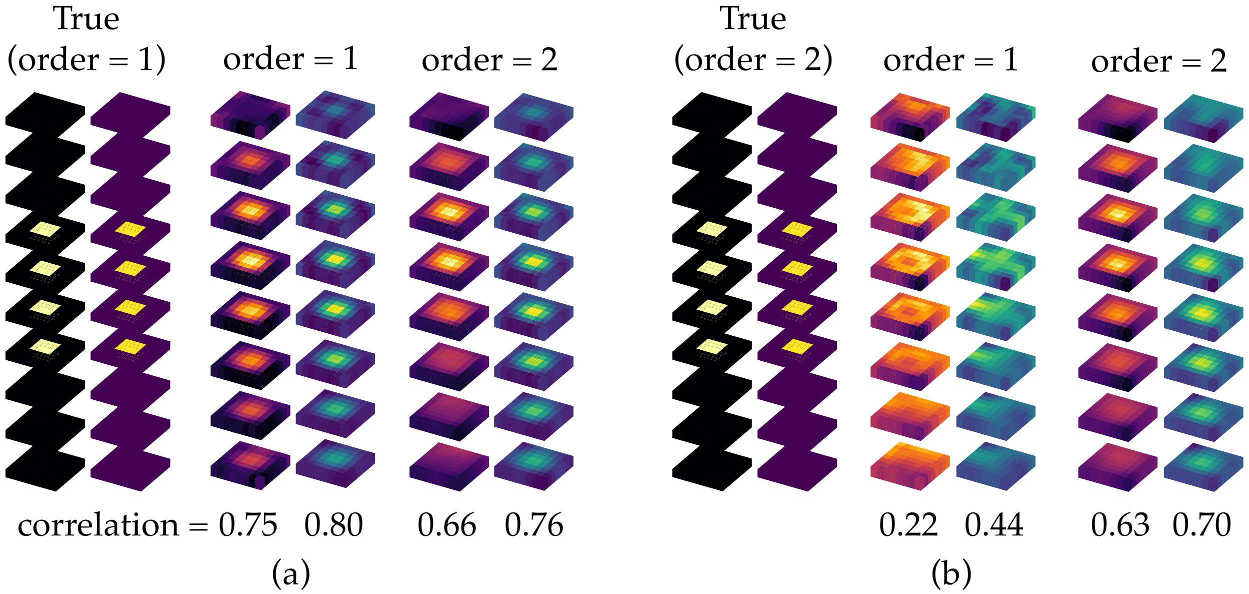
Figure 5.15: The effect of assuming a homogeneous Poisson's ratio. In the true cube, the main material has a Poisson's ratio of $\nu = 0.45$, while the defect material has a Poisson's ratio of 0.3 (roughly corresponding to the values for Jello and clay, resp.). When inferring material properties, we assume a uniform Poisson's ratio across the entire cube. We find that this assumption does not hurt the reconstruction much, especially when ν is set to the Poisson's ratio of the main material. Both estimations use the same 20 motion-extracted image-space modes.

5.10.4.3 Video Capture

Our camera was a Phantom V1610 high-speed camera. Each video was captured at 6000 FPS at an image resolution of 288 × 384. To reduce noise, we averaged every two frames for a resulting temporal frequency of 3000 FPS. Note that in Fig. 9 in the main material, the drums vibrate at frequencies below 120 Hz. While we chose to first demonstrate our approach using a high-speed camera, where compression and camera noise are less challenging, many modal frequencies can be captured on a consumer camera.

5.10.4.4 Extracting Image-Space Modes

We found that in real videos, some level of manual selection was necessary to verify peaks in the motion amplitude spectrum as modal motion. For instance, spurious camera motion would often appear as spikes in the spectrum. Verification was done by visually inspecting the magnified motion in the video at the frequency in question (following the method proposed in [195]). We believe that in the future this step could be automated. The number of extracted modes ranged from 12 to 31, depending on the video.



(a) (b)

Figure 5.16: Element order mismatch. As (a) shows, when using linear elements in the forward model, any element order ≥ 1 suffices in the inference model. However, in (b), reconstruction quality degrades when attempting to model a quadratic-element cube with linear elements.

5.10.4.5 Inference Details

We modeled each drum as a triangular membrane mesh with 1530 linear elements and inferred material properties on a 20x20 pixel grid. In the presented results, the hyperparameter values are $\alpha_u = 10^{12}$, $\eta = 1$, $\alpha_w = 0.1$, $\alpha_v = 0.1$, and $\bar{w} = 10^6$. w and v are initialized to uniform values of 10^6 [Pa] and 10^3 [kg/m3], respectively, and reflect the estimated stiffness and density of latex.

5.10.5 Jello Cube Experiment

5.10.5.1 Inference Details

Our inference model was a 10x10x10 hexadral mesh with linear elements. The optimization hyperparameters were $\alpha_u = 0.1, \eta = 1, \alpha_w = 10^{-10}, \alpha_v = 10^{-8}$, and $\bar{w} = 10000$. w and v were initialized to 10000 [Pa] and 1500 [kg/m3], which are the estimated Young's modulus and measured density values of Jello.

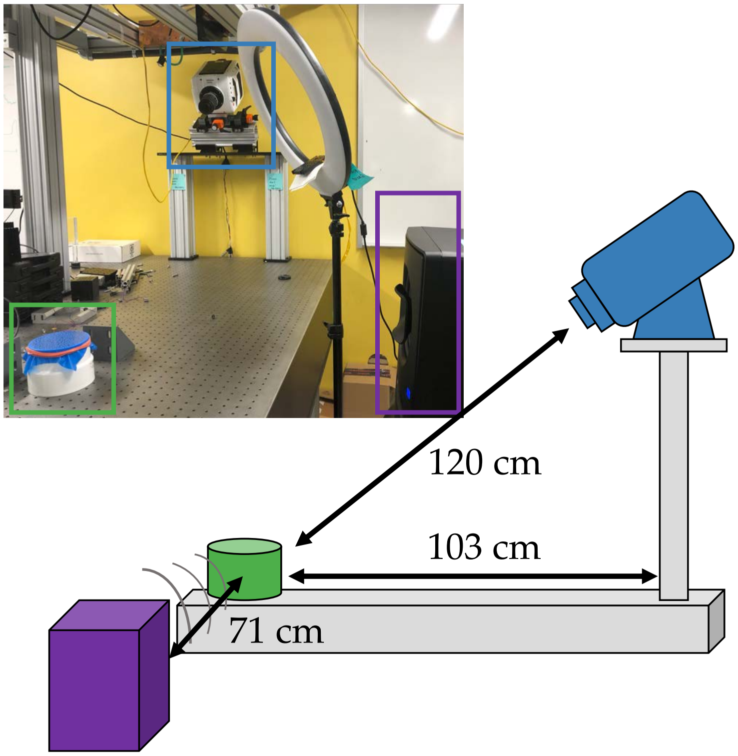


Figure 5.17: Experimental setup for real drums. Vibrations were induced by a loudspeaker and recorded with a high-speed camera.

Chapter 6

VISUAL SURFACE WAVE TOMOGRAPHY: INFERRING SUBSURFACE MATERIAL PROPERTIES FROM MONOCULAR VIDEO

[1] Alexander C. Ogren et al. "Visual Surface Wave Tomography: Inferring Subsurface Material Properties from Monocular Video". In: *preparation* (2024).

Author contributions

A.C.O. conceptualized the project, wrote and revised the manuscript, wrote the simulation code, developed COMSOL models for simulated experiments, and developed and tested the method. B.T.F. wrote and operated the motion extraction code. J.A. assisted with construction of experimental setup, and led the rheometry experiments.

Chapter preface

This chapter continues with the theme of *inexpensive characterization of elastodynamic systems*. Surface wave dynamics can subtley communicate information about the subsurface features and constitution of material systems. Conveniently, surface waves are also the most visibly apparent wave dynamics of most systems. In the interest of developing inexpensive at-home methods to characterize the mechanical stiffness of tissue, an important biomarker for health tracking, the work presented in this chapter Inspired by *Visual Vibration Tomography* (chapter 5), which imaged the interior heterogeneous mechanical properties of objects by observing partial information about their modes of vibration with a monocular video, *Visual Surface Wave Tomography* (VSWT) characterizes layered material systems via the partial observation of surface wave dynamics through monocular video. By inferring the dispersive characteristics of the system from the surface displacements, VSWT uses optimization to converge on a physics-based model that best explains the observations, yielding an inference of the constitutive and geometric parameters of the systems layers.

6.1 Abstract

The mechanical characterization of biological materials is crucial for medical diagnosis and preventative care, real-time imaging during medical procedures, and fundamental science surrounding the modeling of biomechanical systems. To date, the existing techniques to mechanically characterize and image biological materials are ultrasound-based technologies. These methods are quite

impressive with what they can achieve, but because they rely on expensive ultrasonic equipment and highly trained experts to operate, they are often inaccessible outside of medical centers. Numerous studies have demonstrated the importance of early diagnosis and preventative care [211, 212]. This research aims to increase the accessibility of inexpensive, at-home screening methods for the early detection of detrimental health conditions, and thereby improve the overall patient outcome and healthcare experience. In layered wave media, the behavior of surface waves can be strongly affected by subsurface features [39]. Because of this, a principled examination of the surface waves of a system can reveal rich information about the geometry and material properties of features below the surface [213–215]. Further, it has been shown by previous works such as Visual Vibration Tomography [216] that even partial observations of excitations on the surfaces of dynamic systems measured by common equipment such as cameras can provide enough information to form a hypothesis about the object's interior. In this work, we present an inexpensive method to mechanically characterize layered wave systems, such as biological systems containing skin, muscle, fat, and bone. The method uses low frequency excitations to incite surface waves, and observes the system response through monocular video. Using this partial information, computational imaging and inference techniques are employed to form a hypothesis about the mechanical properties governing the system.

6.2 Introduction

Wave-based imaging techniques are used across a wide variety of disciplines as a way to non-destructively infer characteristics about the interior of an object, unseen features in an environment, or other information that would be otherwise unobservable. These methods usually rely on an understanding of how the characteristics of the object or environment affect wave propagation, and align a physics-based hypothesis to fit the observed signal. For example, echo-based navigation systems like RADAR and SONAR assume a constant wave speed, and estimate the distance to objects based on the time it takes for a wave to travel to and from an object. MRI imaging relies on an understanding of how radiofrequency waves will be absorbed and re-emitted by different types of tissues in the body. Based on this understanding, highly detailed images of interiors of our bodies can be reconstructed for diagnostics and biological science. Ultrasonic imaging techniques leverage phased-array transducers to send and receive high-frequency mechanical excitations, processing the reflected signals to triangulate the positions and reflectivity of subsurface features. In some cases, such as Shear Wave Elastography, ultrasonic imaging can even be used to characterize the wavespeeds, and therefore elasticity, of biological features such as tissues and organs.

Medical imaging techniques that characterize the mechanical properties of tissue, such as elasticity, are highly useful for disease detection. For example, tumors often stiffen the tissues they inhabit [40, 41], while degenerative diseases like osteoarthritis [55] and muscular distrophy [56, 57] lead to the softening of cartilage and muscle tissue respectively. Liver diseases such as fibrosis and cirrhosis are characterized by a progressive stiffening of the liver [42–44]. In the context of cardiovascular health, hypertension [50–52] and atherosclerosis [53, 54] can be indicated by an increase in stiffness of arterial walls. Pulmonary diseases such as pulmonary fibrosis [45–47] and chronic obstructive pulmonary disease (COPD) (bronchitis and emphysema) [48, 49] alter the mechanical properties of lung tissue, affecting its ability to expand and contract. Ophthalmic conditions such as glaucoma can be characterized by a change in stiffness of the optic nerve and the cornea [217]. For people with aneurysms, the risk of rupture can often be assessed by understanding the mechanical state of the vessel wall [218], and can be a critically important piece of information to guide surgical interventions.

For many of these conditions, ultrasonic or magnetic resonance approaches are used for diagnosis, such as Transient Elastography [10], Shear Wave Elastography [11], and Magnetic Resonance Elastography [12]. For ophthalmic conditions, optical methods including Optical Coherence Tomography [219] and Optical Coherence Elastography [220] are used. Overall, current methods to detect these conditions often require not only high-end, expensive imaging equipment, but also highly trained medical specialists to operate and maintain the equipment and to interpret the results [221, 222]. These factors make regular screening of patients infeasible, making early detection and disease prevention difficult.

There are many signs to indicate that our society is interested in making at-home health monitoring more accessible, and more advanced. Wearable devices initially began as fitness trackers that would count steps or provide GPS tracking of exercise activities. Eventually, these devices began to see the integration of heart rate monitoring, allowing a more accurate inference of exercise intensity, and for the first time, allowing quantitative and affordable at-home monitoring of sleep quality and stress level. Recently, we've seen wearable devices capable of SpO_2 and VO_2 max measurements, temperature measurements, and even basic ECG capabilities. As wearable sensor technology advances and large amounts of biometric data become increasingly available, our ability to leverage this data for health inference has significantly improved. These inferences can range from early disease detection, to physical rehabilitation tracking, metabolism tracking, and menstrual tracking.

While many medical imaging techniques use bulk waves (e.g. shear waves and pressure waves for ultrasound, and electromagnetic waves for magnetic resonance), surface waves can also be highly useful for subsurface imaging. Surface waves are waves that travel predominantly along the surface boundary of a medium. They are studied across a wide span of physical sciences, including fluid dynamics, solid mechanics, optics, atmospheric science, and geophysical and seismic science. In the context of geophysics and siesmology imaging, surface waves are used to infer constitutive and geometric information about strata (the geological layers making up the Earth) [6–8, 39, 213–215].

This is possible because although surface waves require a surface boundary in order to exist, their dynamics extend to some depth below the surface. Because of this, they can be highly sensitive to features and variations in material properties below the surface. Because these techniques only require data from the surface of the medium, and do not require comprehensive scans (which often require phased-arrays), surface-wave-based imaging techniques often require less equipment and operating cost than their bulk-wave counterparts. Although they often are constrained by more limitations that bulk-wave based imaging technology, surface-wave-based imaging methods offer an inexpensive alternative to provide quick and approximate characterizations of subsurface materials and features. Geophysical techniques that utilize surface wave physics include Seismic Refraction and Reflection [5], Multichannel Analysis of Surface Waves (MASW) [6], Ambient Noise Tomography [7], Spectral Analysis of Surface Waves (SASW) [8], and more generally, Surface Wave Dispersion, Rayleigh Wave Analysis, and Love Wave Analysis. Of specific relevance to my work, MASW and SASW analyze the spatiotemporal spectral properties (in other words, the dispersion relation) of surface waves to infer the stiffnesses and thicknesses of subsurface strata. Of interest for future directions of this work, Ambient Noise Tomography utilizes continuous, low-level background seismic noise to construct images of the Earth's subsurface, without the need for externally applied excitations.

Interestingly, these types of surface-wave-based techniques have not been extensively studied in the field of medical imaging. From the well-established surface wave methods in seismology, it is clear that surface waves carry rich subsurface information. From Visual Vibration Tomography, we know that we can leverage partial surface information obtained via camera to infer the interior properties of dynamic systems. Visual Surface Wave Tomography combines these ideas, leveraging the observation of surface waves via monocular video to infer the properties of the subsurface strata. It is our hope that this research will advance the field of at-home, routine biometric inference via generally affordable technology (e.g. smartphone cameras or wearables) to enable the next generation of accessible health tracking for early detection and preventative care.

6.3 Problem setting

The ultimate goal of this research is to use monocular observations of surface wave displacements to fully characterize the strata of an elastic layered system. This characterization includes a hypothesis of the elastic constants governing each layer, and the thicknesses of each layer. As a first approach to this problem, we define a relatively simple system consisting of two layers of elastic material. The first layer will represent bone, and the surface layer will represent soft tissue (such as skin, muscle, or fat). In comparison to the soft tissue, the bone material will be nearly rigid (at least three orders of magnitude stiffer than the soft tissue layer). For this system, we will aim to estimate the thickness and elastic modulus of the soft tissue.

6.4 Methods

In this section, we delineate the methods by which we carry out this inference. The pipeline involves the following steps:

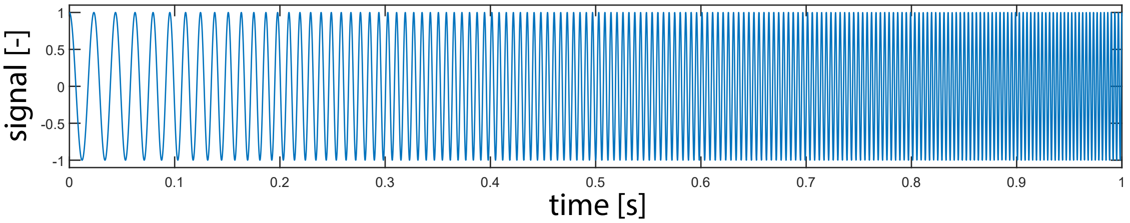
1. Provide an excitation to the layered system to incite surface waves.
2. Observe these surface waves with a camera.
3. Use phase-based motion extraction to obtain the surface displacements.
4. Extract the dispersion relation of the surface waves via FFT of the surface displacements.
5. Infer the elastic modulus and thickness of the soft tissue by fitting a physics-based model to match the observed dispersion relation.

These steps can be roughly broken up into experimental and computational methods. Experimental methods involve the techniques used to excite the system and observe the response. Computational methods involve the processing of the observed signal, and the inference of the underlying system parameters from that data.

6.4.1 Exciting surface waves

This method fundamentally relies on being able to observe surface waves as the system responds to an excitation. To excite surface waves in our system, we prescibe a vertical excitation on the surface of the medium. In real-world experiments, we apply the excitation through the use of a shaker. We aim to excite waves over a range of frequencies, allowing us to observe a more comprehensive realization of the dispersion relation. To do this, we use a *chirp* excitation, which is a sinusoidal signal that has a continuously time-varying frequency. These types of excitations are commonly used to provide information about the response of a system over a frequency range in a much more efficient manner than a discrete frequency sweep, where the excitation would consist of a series of excitations at single fixed frequencies. In the context of real-world experiments, this is incredibly important, because we need to be able to capture information about many frequencies in a short amount of time such that they can be reasonably captured on video in a way that isn't excessively demanding in terms of file storage or processing. For real-world applications, it would be important to extend this work to examine the effect of using simpler excitation methods, such as impulses, since frequency-controlled excitations may not be available in common settings, and a major pillar of this work is to be accessible without expensive equipment. Alternatively, it may be feasible to generate a chirp using a smartwatch haptic engine.

(a) chirp signal



(b) spectrogram of chirp signal

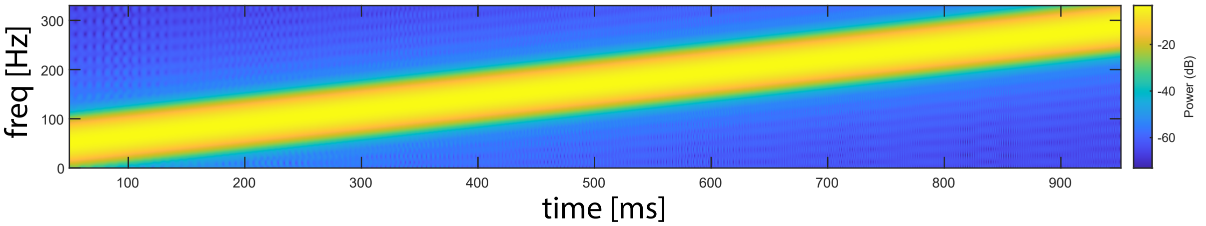


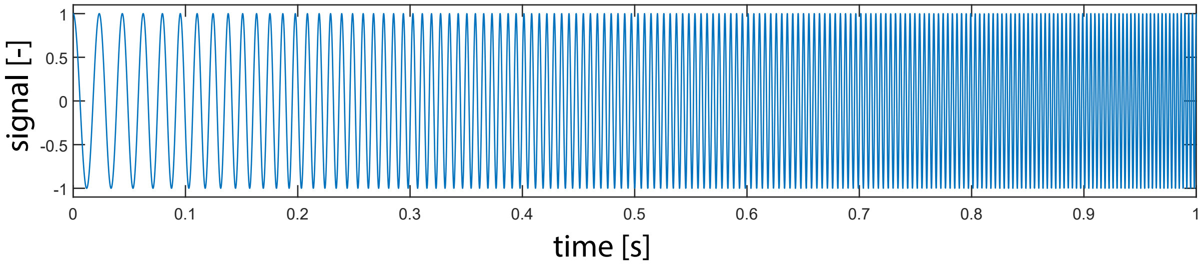
Figure 6.1: An example of a chirp excitation with a gradual variation in amplitude, along with a spectrogram showing the time-varying frequency content of the chirp. This particular chirp sweeps from 40 Hz to 300 Hz

In practice, to ensure that the system does not experiences impulses (which may cause unexpected behavior both numerically and experimentally), we make sure to turn the chirp on and off gradually. We use a smoothed heaviside function with a continuous second derivative ($flc2hs$, credit to Comsol Multiphysics) to achieve this, whose equation is given in Eq. 6.1 below, and demonstrated graphically in Fig. 6.2. It is important that this function specifically has a continuous *second* derivative, because this means that any inertial forces (which are related to acceleration, the *second* derivative of displacement) caused by the excitation will change in a continuous manner.

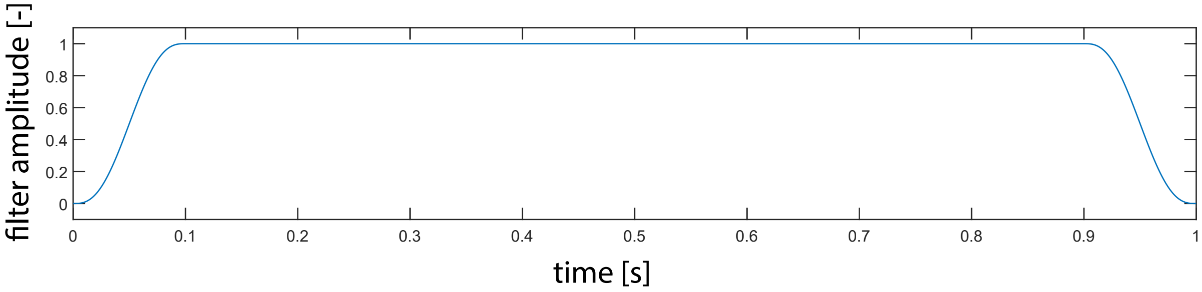
$$flc2hs(x,d) = \begin{cases} 0, & \text{if } x \leq -d \\ 1, & \text{if } x \geq d \\ 0.5 + 0.9375 \cdot (\frac{x}{d}) - 0.625 \cdot (\frac{x}{d})^3 + 0.1875 \cdot (\frac{x}{d})^5, & \text{if } -d < x < d \end{cases} \tag{6.1}$$

For this study, we assume the excited waves will be plane waves. Because we aim to initially attack this problem from the simplified two-dimensional standpoint, the numerical model implementations involved in this research are currently based on a plane-strain assumption. For real-world applications, it would be important to extend this work to three-dimensional geometry, since excitations in real objects like the body are unlikely to follow two-dimensional assumptions such as plane-strain.

(a) original chirp



(b) smoothed heaviside with continuous second derivative (flc2hs)



(c) chirp with smoothed heaviside filter applied

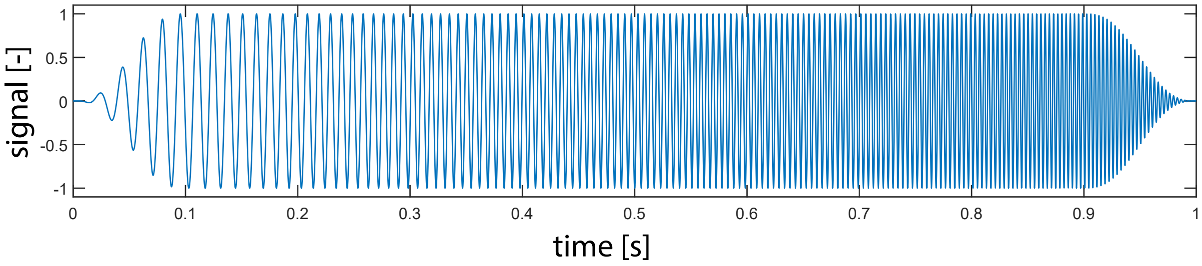


Figure 6.2: Chirp excitation with smoothed heaviside filter.

6.4.2 Extracting displacements

After exciting the system, we observe the surface waves supported by the system, which carry rich information about the medium they are traveling on. In real-world experiments, we do this by observing the system with a camera, and applying phase-based motion extraction to the video [223]. Note that displacements at depth are not available, since the camera cannot see below the surface of the system. The result is a series of frames that show surface displacements in image space, both vertically and horizontally. Image space is the two-dimensional projection of the three-dimensional physical world onto the sensor of the camera. For clarity, these directions in image space will be referred to as *image-space horizontal* and *image-space vertical*, for example, to differentiate from directions in three-dimensions. Currently, we orient the camera such that the plane waves travel directly in the image-space horizontal direction. Conveniently, this allows us to approximately ignore image space to physical space projections, and still get reasonable results. For real-world applications, it would be important to more precisely account for this projection, since observations are not always guaranteed the convenient orientations that we have engineered into our experiments.

6.4.3 Extracting dispersion relations

After surface displacements are computed, we convert them into information that more directly and concisely describe how waves travel in the layered medium. We estimate the dispersion relation from the displacements by taking their Fourier transform over both space and time. The dispersion relation, therefore, encapsulates the relationship between the spatial and temporal periodiocity of the waves supported by the layered medium. The information contained in the dispersion relation is fundamentally the information that we are using to make an inference as to properties of the strata.

6.4.4 Inference problem

With the observed dispersion relation of the system characterized, we can use this information to infer the properties of the strata. To do this, we employ a Bloch model representing a layered wave medium, and try to determine the optimal strata properties used in the Bloch model that best agree with the observed data. We define the following score function, and aim to find the system parameters that maximize it. The score function is given by

$$E = \sum_{i=1}^{N} \int_{0}^{1} f\left(\vec{c}_i(s)\right) ds \tag{6.2}$$

where E represents the score, $f(\vec{x})$ gives the power of the observed dispersion relation for any wavevector-frequency pair $\vec{x}$, and $\vec{c}_i(s)$ parameterizes the bands of the dispersion relation found by the Bloch model, for contour parameter $s \in [0, 1]$. Put simply, the score increases when the bands of the Bloch model overlay with regions of high power in the observed dispersion relation, and decreases when this agreement does not happen. This score function rewards good agreement between the observed data and the physical model.

In the current implementation, the fitting algorithm we use is a comprehensive grid search. Although this solution is quite heavy-handed and wasteful, it is indispensable in the research-and-development phase in that it can accurately reveal the entire score landscape without guesswork. In real-world applications of this research, it would be important to determine more computationally efficient ways to perform this fitting problem. Ideally, the fitting routing would be able to run quickly on low-powered devices such as smartphones, smartwatches, or other wearable devices. In the absence of a more elegant fitting algorithm, a real-world realization of this technology could also rely on off-device computing.

6.4.4.1 Bloch model

The Bloch model we employ uses in-house code library, `tissue-dispersion`, for the hyper-efficient calculation of the numerical dispersion relation (`https://github.com/aco8ogren/tissue-dispersion`). Efficiency is key when using this model, because there is a need for repeated evaluation of the model during the fitting problem, especially when paired with the comprehensive grid search fitting algorithm previously described. The model attains this efficiency by using intelligent meshing (regions that are not expected to have high strain gradients do not need fine meshes), cleverly avoiding quadrature calculations, and circumventing the overhead that would typically be associated with using non-specialized, commercially available FEM packages.

Fundamentally, the model uses the finite element method with linear quadrilateral elements. The mesh is designed entirely with rectangular elements. This construction enables us to precompute the element stiffness matrices as a function of their size and constitution, allowing us to entirely circumvent the computational costs of quadrature (which is typically dependent on an arbitrary element geometry) in the assembly of the mass and stiffness matrices, leading to significant computational savings. Further, `tissue-dispersion` employs efficient data typing (e.g. sparse matrices), vectorized assembly of system matrices, parallelized eigenvalue computations, and an efficient, symmetry-aware eigenvalue solver.

6.5 Results

6.5.1 Numerical experiments

In this section, we demonstrate the use of our method on numerically simulated systems. This study provides an evaluation of the method in a noise-controlled and assumption-controlled environment.

6.5.1.1 Scenarios

We examine 5 scenarios of varying thickness to assess whether the method can differentiate between these cases. For each case, the elastic modulus of the soft region (upper layer) is 5 kPa, while that of the stiff region (lower layer) is 100 MPa. This is a $20,000\times$ difference in elastic modulus, representative of the stark difference between the low modulus of soft tissue and the high modulus of the underlying bone. The thickness of the bone layer in each scenario is 30 mm. The thicknesses of the soft tissue in each scenario are th_s = 5 mm, 10 mm, 20 mm, 30 mm, and 40 mm. The density of both layers is 1000 kg/m^3 for all scenarios. The poisson ratio of both layers is 0.45 for all scenarios.

6.5.1.2 Transient models

As a stand-in for the observation of real-world materials, we numerically simulate layered systems using the finite element method in Comsol Multiphysics, a commercial software for physical simulation. In these simulations, we employ transient models so that we can directly simulate how a layered wave system would response to a surface excitation over time. In these numerical experiments, excitations are provided by simply enforcing a time-dependent dirichlet boundary condition on a small patch of the surface of the domain, defined by a smoothed chirp excitation from 40 [Hz] to 225 [Hz]. A diagram showing the layout of the transient model for the 20 mm thickness case is shown in Fig. 6.3 below.

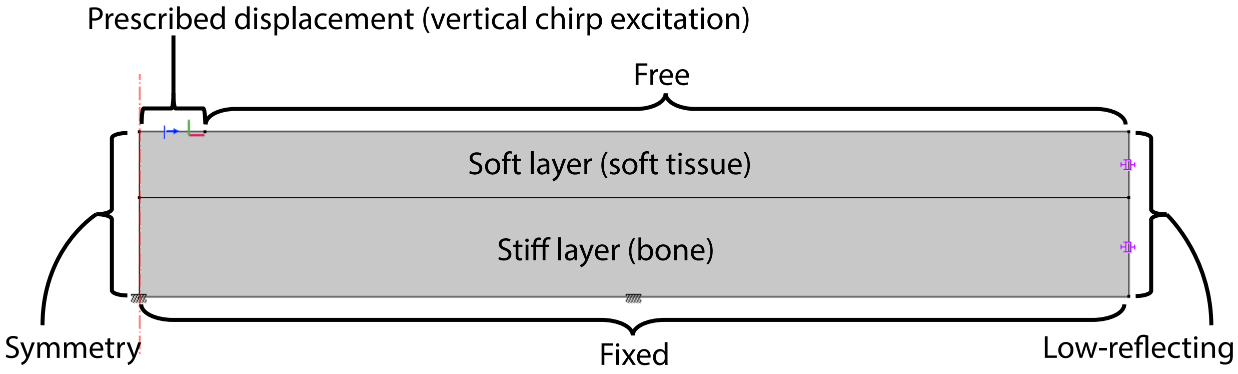


Figure 6.3: A diagram showing the geometry and boundary conditions of the transient model for the 20 mm case.

The full domain for each transient simulation measures 30 cm horizontally, and its height is defined by the sum of the thickness of the stiff layer and the soft layer. Temporally, each simulation spans 0.5 seconds and records displacements at 1000 timesteps. In Comsol, the time-stepping scheme is set to 'strict', meaning that Comsol will compute the solution at each of the 1000 timesteps where the solution is recorded, but is also allowed to take additional time steps in order to satisfy convergence tolerance parameters, for example. This ensures that the time resolution is sufficient to resolve the chirp excitation, since a measurement rate of 1000 samples/sec means the Nyquist frequency is 500 Hz, well above the maximum frequency of 225 Hz provided by the chirp.

The mesh of the transient model uses quadrilateral linear elements. We resolve the mesh finely enough such that even the highest frequency content included in the chirp will be well represented by the span of the element shape functions. The mesh resolution is controlled such that the element size in the soft layer is identical across all scenarios. The side length of each element in the soft region is 5×10^{-4} m across all scenarios, about 10× smaller than any wavelength induced by the chirp excitation (using Rayleigh surface wave solution to estimate the frequency-wavelength relationship). The stiff layer always has the same mesh across all scenarios, since its thickness does not change. We use a coarse mesh for the stiff layer to save on computational cost since large gradients in the solution are not expected in this region. A zoomed in (and therefore horizontally truncated) view of this mesh for the 5 mm case is shown in Fig. 6.4 below.

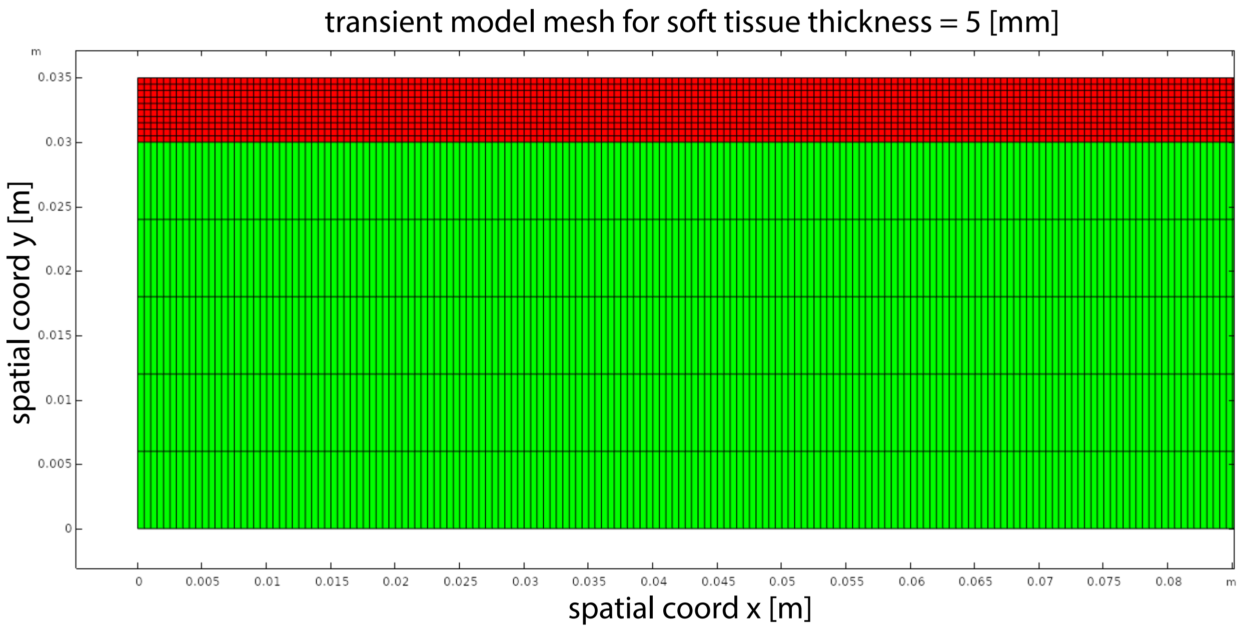

transient model mesh for soft tissue thickness = 5 [mm]

Figure 6.4: A zoomed in (and horizontally truncated) view of the mesh used for the transient model representing the 5 mm case.

6.5.1.3 Mesh controls to match discretization between Bloch models and transient models

In any simulation, physics is being only approximately represented. As the discretization used by a simulation is refined, we often expect the solution to converge to an analytically correct solution. Practically, this means that models using different discretization schemes may yield slightly different results. To encourage greater agreement between the Bloch model and the transient model, we carefully control the discretization of each model so that their meshes will capture approximately the same physics. To achieve this we simply create the Bloch model mesh such that the element size is the same as that used in the transient model. Because the Bloch model is often evaluated with soft tissue thickness parameters that are not integer multiples of the transient model element size, we pick the Bloch model element size that is closest to matching. The calculations to achieve this are given in Eqs. 6.4 below.

$$N_{\text{ele,Bloch}} = \text{round}\left(\frac{th_{\text{s,Bloch}}}{h_{\text{ele,transient}}}\right) \tag{6.3}$$

$$h_{\text{ele,Bloch}} = th_{\text{s,Bloch}} \cdot N_{\text{ele,Bloch}} \tag{6.4}$$

6.5.1.4 Displacements

In this section, we show the displacement profiles for each scenario. Surface displacements are extracted by querying the finite element solution at an evenly spaced array of points along the surface boundary of the domain at each point in time. We measure displacements only along the surface because this is the only information we will have access to in real-world applications. Displacements at depth are not visible to the camera. There is no noise associated with these displacement measurements. Visualizations of these displacements can be found in Fig. 6.5.

6.5.1.5 Observed dispersion relations

We apply FFT in each direction to extract the dispersion relation from the surface displacements. It should be noted that because we are only measuring surface displacements, only modes that participate in surface motion will show up in this measurement of the dispersion relation. That means that modes that are primarily bulk wave modes may exhibit much lower power in the FFT, or may not show up at all. These dispersion relations are shown in Fig. 6.5 alongside their corresponding displacements. It is evident that the dispersion relation is more sensitive to the thickness of the soft layer for shallower thicknesses. This is entirely expected, since surface waves are only affected by subsurface features up to certain depths, often related to the wavelength of the wave.

6.5.1.6 Inference: fitting a physics-based model to observed data

To infer the underlying parameters that define the strata of the observed wave medium (specifically the thickness and elastic modulus of the soft tissue layer), we find the Bloch model that best fits the observed data. Using an exhaustive grid search, we comprehensively sweep over a region of parameter space around the true parameters, evaluating the Bloch dispersion model for each set of parameters, and assessing its score against the observed data using Eq. 6.2. For this study, the resolution of the grid search is 80×80. The Bloch model computes 6 eigenvalue bands.

The results for the inference problem are shown in Fig. 6.6 for each scenario. Note that the inference problem works fantastically for each of the cases from 5 mm to 30 mm, but seems to break down for the 40 mm case. We find that the mechanism for this that a Bloch model with very high thickness will have bands that are very close together, and can all overlap with the same "hot spot" in the observed dispersion relation. This is an unintended behavior of the current scoring function, and is a good candidate for improvement in a future version of this research. For each scenario, comparisons of the Bloch dispersion relation overlayed on the observed dispersion relation for both the true strata parameters and the inferred strata parameters in Fig. 6.7.

It is somewhat unsurprising that systems with greater layer thickness are more difficult to characterize. Firstly, surface waves are only affected by subsurface strata *up to a certain depth* below the surface. This is because most of the dynamics of these modes are concentrated near the surface, and extend to a depth below the surface roughly proportional to the wavelength of the wave.

6.5.1.7 Sensitivity analysis: how sensitive is this method to small changes in parameters?

It's exciting that this method can accurately estimate the parameters of a simple layered system for parameter values that are far apart. However, we also want to understand how sensitive the method is to tracking small changes in the properties of the strata. To help answer this question, we perform a sensitivity analysis, where we make small changes to the true parameters of the observed system, and repeat the inference problem for each perturbation. Because this requires running a large number of transient simulations (which are expensive), this study is only feasible at a relatively small scale. Further, because we have to run the inference fitting problem many times, we reduce the resolution at which we perform the comprehensive grid search from 80×80 to 20×20 to save on computational cost. For each scenario, we perturb the parameters by 10% in each direction, and perform the inference problem for each of these perturbed parameter values. The results for this study are depicted in Fig. 6.8, showing that VSWT can quite effectively differentiate between systems with parameter changes smaller than 10% perfectly for the 5, 10, and 20 mm cases. The method begins to show some signs of failure, but still performs overall quite well for the 30 mm scenario. As expected from the results of the previous section, the 40 mm case is outside the safe range of operation for the current version of VSWT, showing that for all parameter cases, the method strongly artificially gravitates toward the solutions with the greatest soft tissue thickness.

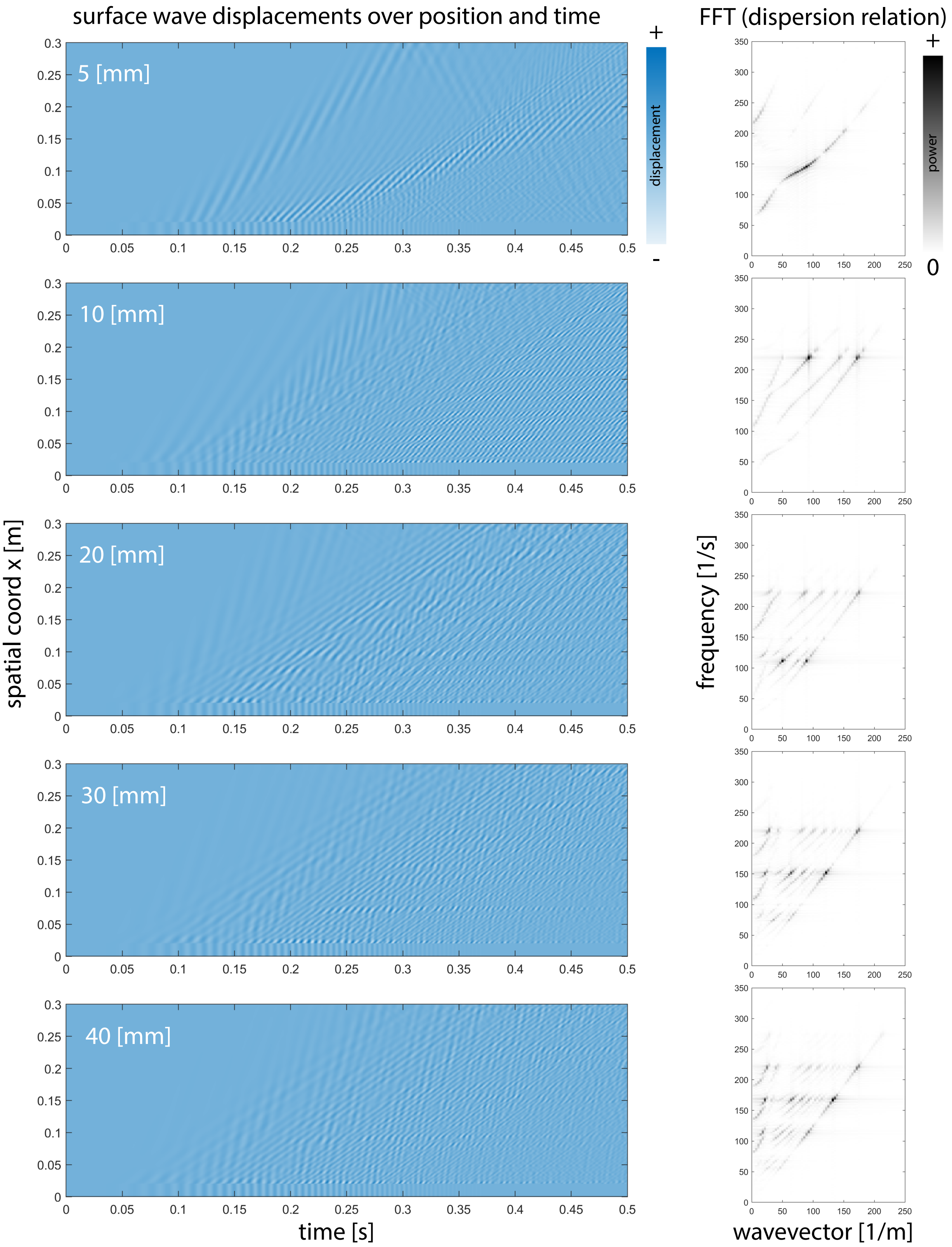


Figure 6.5: For each scenario, the displacements are plotted in the left column with the observed dispersion relation, as computed by FFT, on the right.

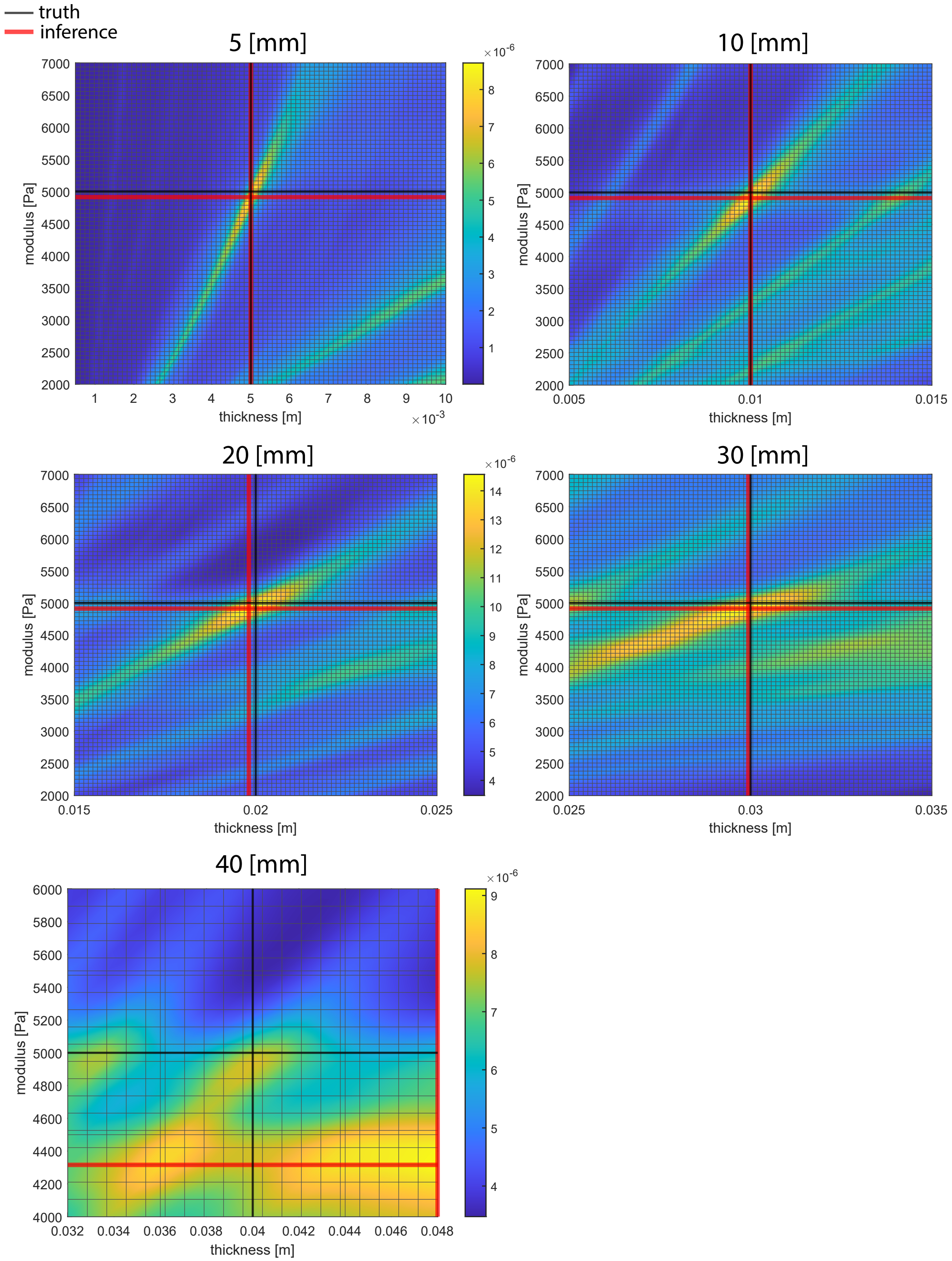


Figure 6.6: Results of inference problem for each of five scenarios with varying soft tissue thickness. The inference problem breaks down for the 40 mm case.

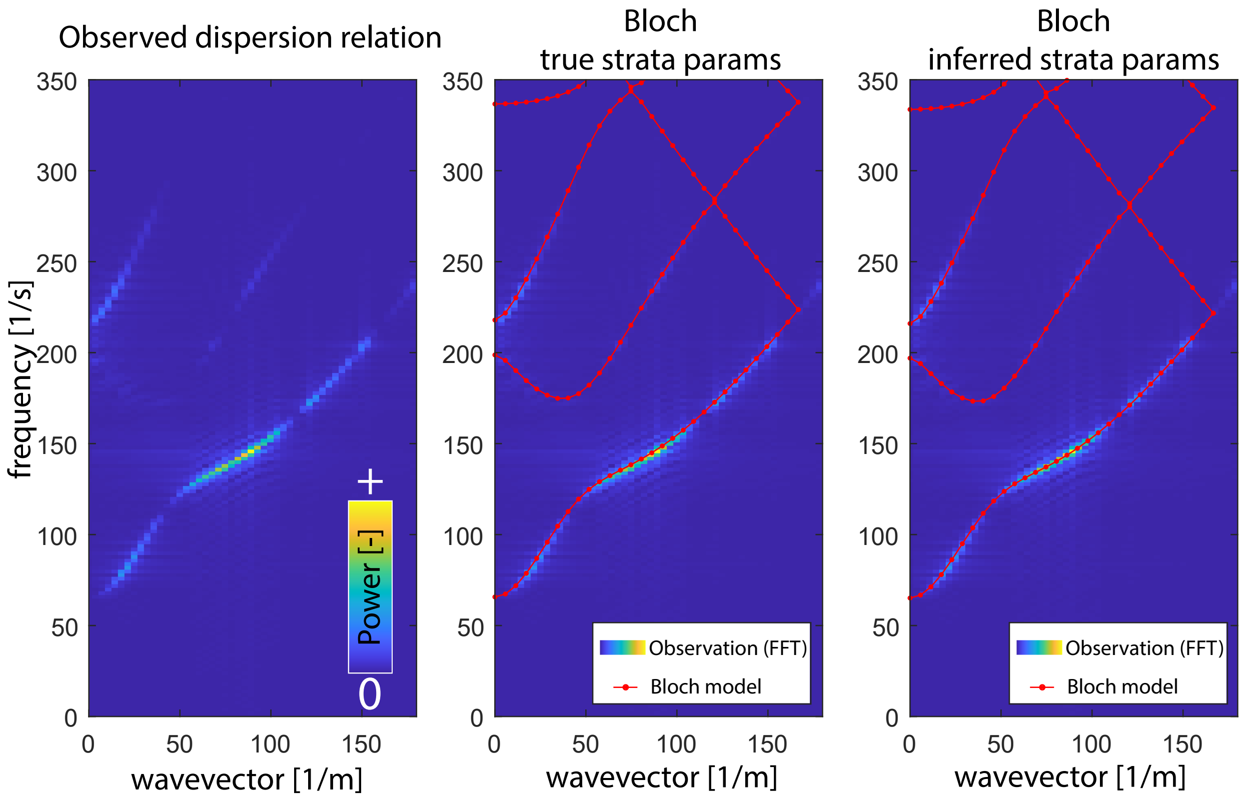


Figure 6.7: A comparison of the dispersion relation as computed by the Bloch model against the observed dispersion relation for both the true strata parameters and the inferred strata parameters for the 5 mm scenario.

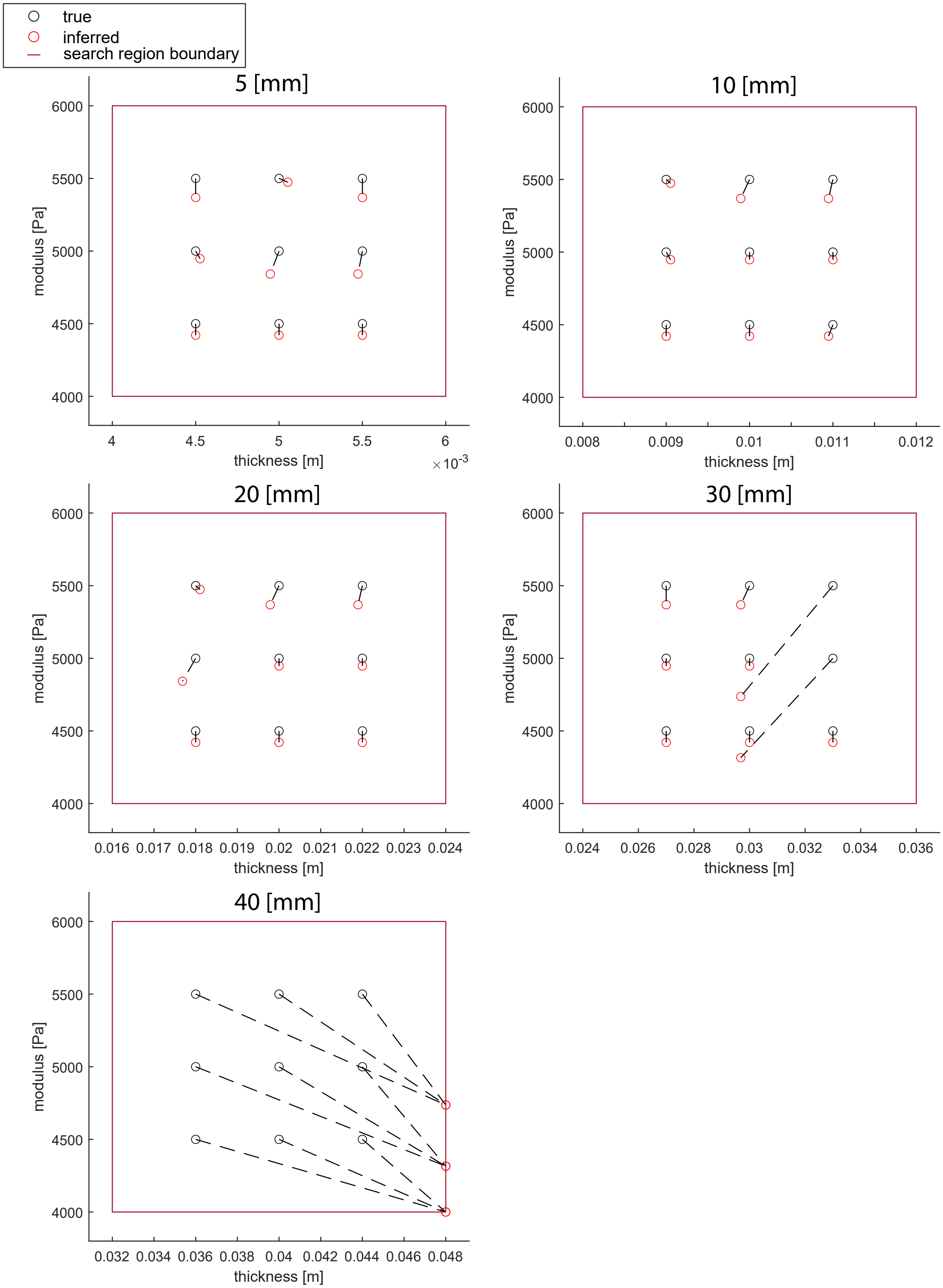


Figure 6.8: Results for the sensitivity analysis of each of the five scenarios.

6.5.2 Real-world experiments

6.5.2.1 Materials

To construct a real life layered system, we use brand name Jell-O (peach flavor). Jell-O is a gelatin based hydrogel commonly used in culinary endeavors for desserts. We pick this material because it has the rare combination of having both relatively low stiffness and low damping, meaning that excitations will be easier to observe. In addition, the mechanics of many tissues in the body are predominantly governed by another hydrogel, the Extracellular Matrix (ECM), based mostly on collagen and elastic fibers [224–228]. Because gelatin-based hydrogels and the ECM share many hydrogel-based characteristics, they are often used as phantom materials in biomechanical studies [229–231]. The per-volume costs of actual biological tissue, or collagen and elastin can be extremely high in comparison to Gelatin.

For the stiff layer below the Jell-O, we use acrylic. Since we set the Jell-O in the acrylic container, we expect the interface between these two materials to be relatively slip-less. Two samples are used. Sample A has an unknown true thickness (the thickness was measured before Jell-O polymerization, and we believe the thickness changed during the setting process). Sample B has a true thickness of 16 mm. Neither sample A nor B has a known elastic modulus. Preliminary rheological experiments have been performed to gain a ground truth measurement of the sample properties, but we are currently skeptical of the results. As a result, the rheological results are not reported here.

6.5.2.2 Excitation

To provide a chirp excitation to the surface of the sample, we use a B and K shaker along with a custom crafted chirp excitation ranging from 40 Hz to 300 Hz. Because this research is currently being done in two-dimensions, we have to modify our three-dimensional experiment to emulate the two-dimensional plane strain conditions. To do this, we use an actuator head that encourages the formation of a plane wave. An image of the actuation geometry is shown below in Fig. 6.9.

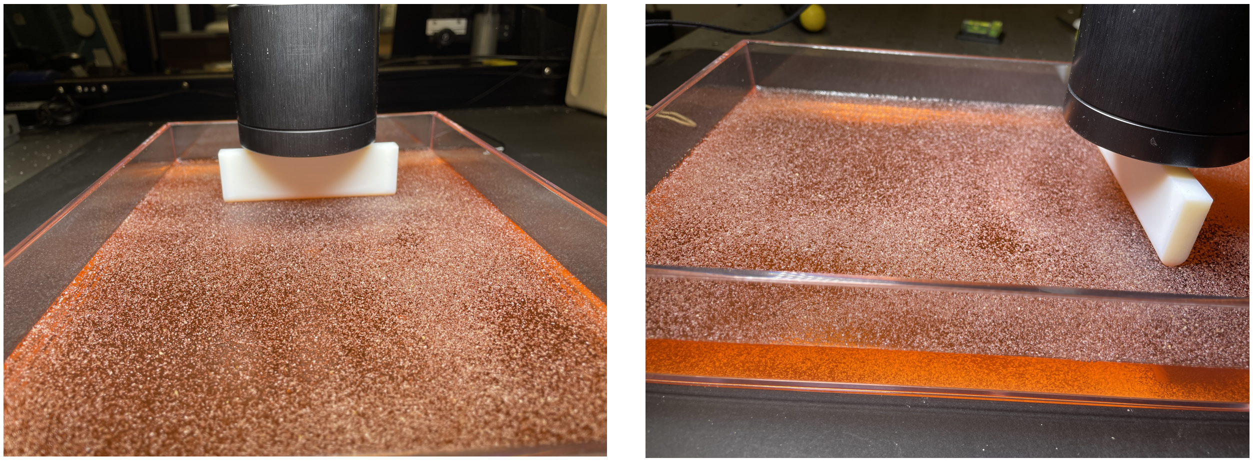


Figure 6.9: Views of the actuation mechanism from two angles. The black cylinder is the B and K shaker. The white attachment is the actuation head, custom designed to encourage the formation of plane waves.

6.5.2.3 Observation: high speed camera

For the current state of this research, we use a high speed camera to observe the surface excitations. We operate the high speed camera at 1000 fps. Because biological materials are soft, they support relatively low frequency waves, and do not require high frequency sensing to detect. However, for now, the use of the high speed camera allows us to more easily develop the method without worrying about confounding effects that could be caused using a less science-oriented camera with a lower framerate and smaller lens.

In the future, we are interested in evaluating this method with more affordable equipment (e.g. smartphone cameras). In addition, although the high speed camera is capable of framerates over $100,000$ fps, we observe our system at only 1000 fps. Ultimately, the observation of surface waves traveling on biological materials should be within the reach of consumer-grade devices such as smartphones and wearables. In fact, the Samsung Galaxy S23 (2023), one of the most popular flagship smartphones, has the capability to capture video at 960 fps, while the Xiaomi 14 Ultra (co-engineered with prominent camera company, Leica) (2024) supports video recording up to 1920 fps.

6.5.2.4 Extracting displacements

To extract displacements, we use phase-based motion extraction [223], which allows us to observe subpixel displacements. This allows us to ensure that the sample stays in the linear elastic regime during excitation, since we do not need to excite superpixel displacements. To improve the quality of motion tracking, we speckle the surface of our sample with garlic powder. An image of the sample is shown in Fig. 6.9. Examples of the extracted displacements are shown in Fig. 6.10.

6.5.2.5 Conversion of image-space displacements to physical displacements

To convert from image-space displacement to physical displacements, a signpost method is used. This means that we measure the physical size of an in-frame object, and use it's pixel measurement to determine the approximate physical size of a pixel, for pixels in the same plane of focus as the signpost. This neglects the fact that the physical size of a pixel is not constant across a frame, even if the plane of focus is kept constant, since objects near the periphery of the frame will be slightly further from the lens than those in the center of the frame. More rigorously accounting for the transformation between image space and physical space is a worthwhile future improvement for this project. However, the current signpost method is sufficient for a proof-of-concept.

Because the camera viewing angle is orthogonal to the direction of wave propagation, we define the basis for our quasi-two-dimensional wave propagation to be aligned with this viewing angle. As a result, horizontal displacements in image space can be approximately equated to horizontal displacements in physical space. Vertical displacements in image space can be approximately considered to be directly proportional to vertical displacements in physical space.

6.5.2.6 Extracting observed dispersion relations

To convert the observed data to a useful format, we apply the same FFT processing as for the numerical experiments. However, we first must convert the signal to have only one spatial dimension and one temporal dimension. To do this, we average the signal across pixels that are in the same column, over a very small window of pixels. Further, we crop the signal spatially and temporally such that the signal only repeats once, and we only observe surface waves on one side of the actuator head, with the actuator head out of frame. The resulting signal is visualized in Fig. 6.11.

After doing this processing, the observed dispersion relation is computed with FFT. Two examples are shown in Fig. 6.12.

6.5.2.7 Inference fit on real observations

After computing the observed dispersion relation via FFT, we attempt an inference fit. Unfortunately, at the current state of this research, the automated fitting routine fails. Among the factors that may contribute to this:

1. The gap between real-world physics (observed) and linear elasticity (Bloch model used for fitting). Hydrogels are real-world materials that only somewhat obey linear elasticity. In

reality, they are fiber-based materials that have complicated physics and are being actively studied today. Of primary importance to this research are the frequency dependent mechanics, and viscoelastic effects.

2. The gap between three-dimensional physics and our two-dimensional Bloch model. There will be three-dimensional modes and effects that are not well represented by our Bloch model that could affect the inference fit.
3. The gap between the true physical displacements that are occuring, and our observation of them. Cameras are subject to noise, and our processing of the data is imperfect. For example, we only approximately account for the conversion between image-space and physical-space. Further, the averaging we perform across pixels may not be the best way to average. Averaging in FFT space (possibly even across multiple observations of excitations) may be a more robust approach.

For these reasons, the converged-upon solution is not accurate. Alternatively we can attempt a manual fit, using our own visual estimation for which Bloch model best fits the observed data. The manual fit is performed with the help of the *tipiX* tool, maintained by Adrian Dalca, details of which can be found at `https://github.com/adalca/tipiX`. This method yields more reasonable results, but it is clear that a more complex physics-based model will be needed in the future to better represent real-world physics of hydrogels. The results of the manual fit are shown in Fig. 6.12.

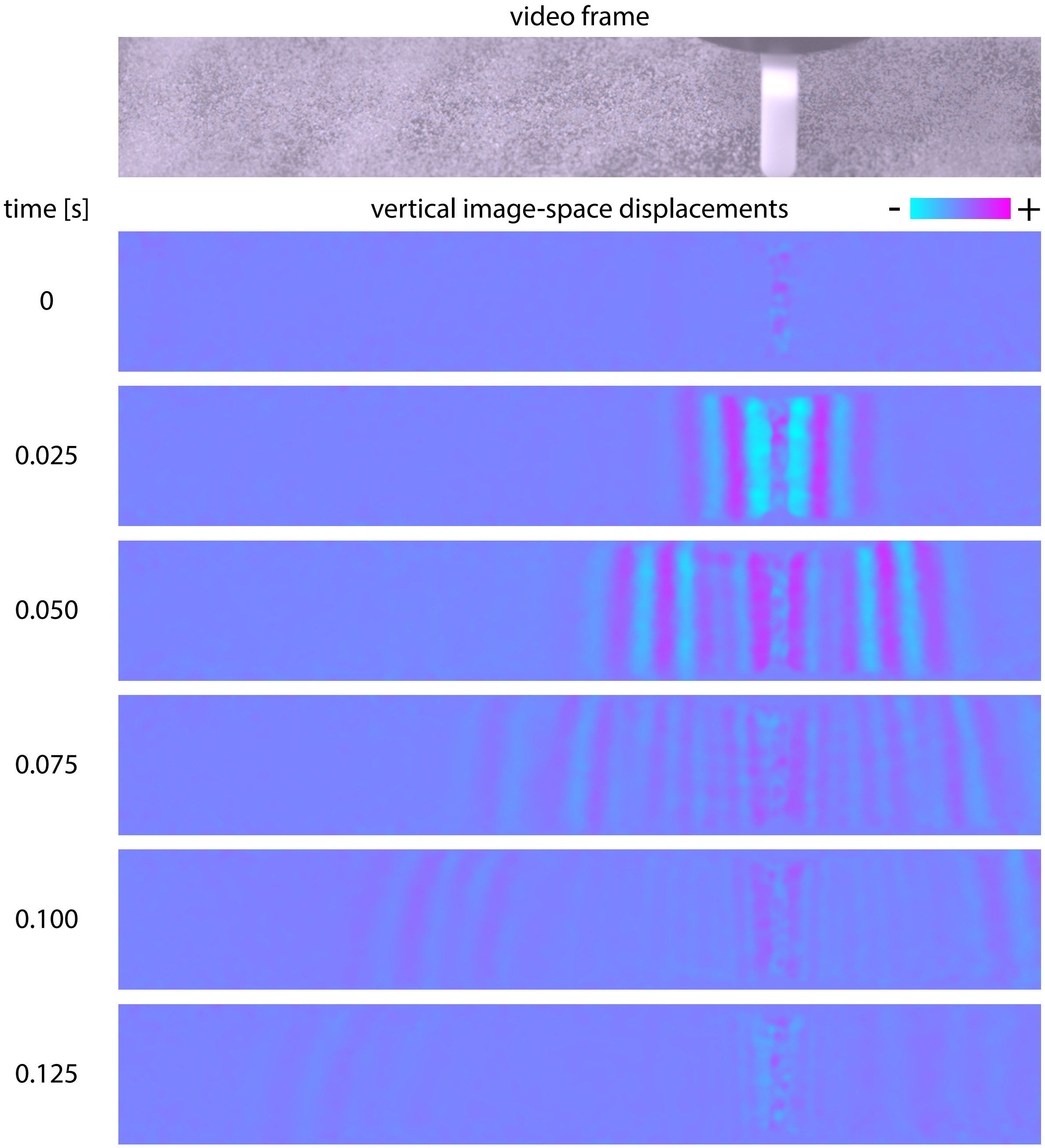


Figure 6.10: Image-space displacements extracted from video. The displacements shown are vertical. Surface waves can be observed traveling from the actuator head.

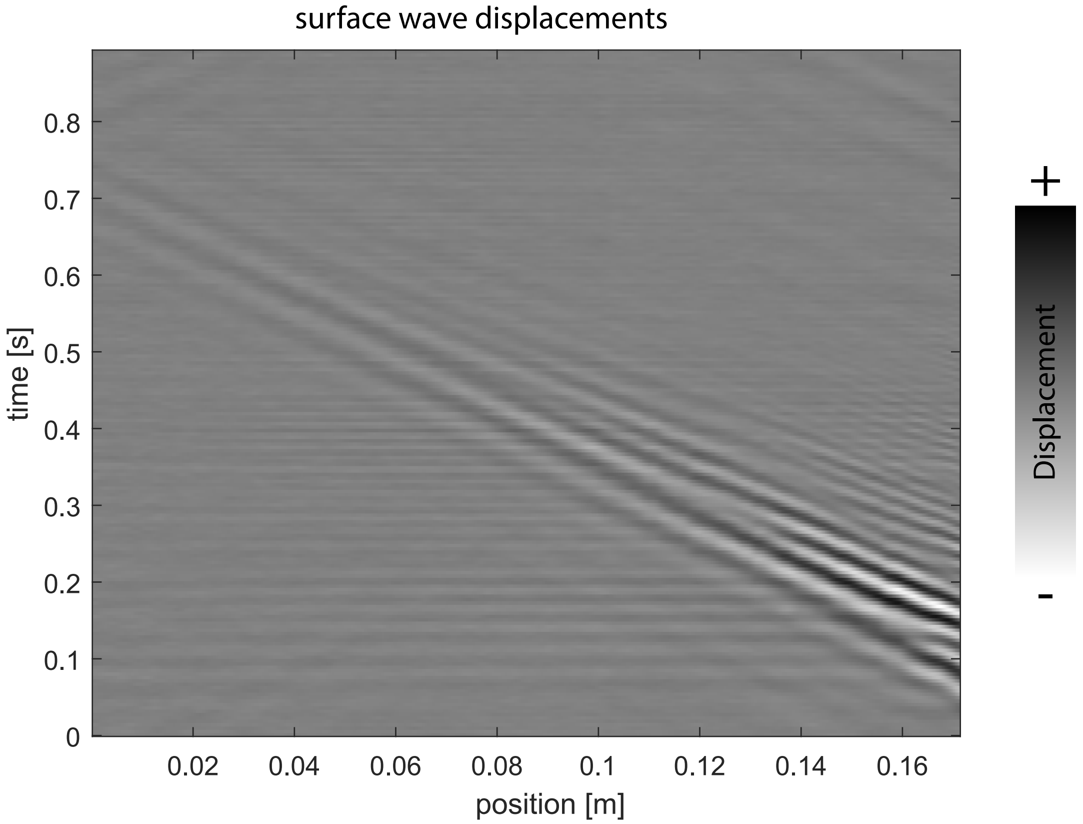


Figure 6.11: Image-space vertical displacements spatially and temporally cropped to show one repetition of the excited surface waves, with the actuator out of frame. Displacements are vertically averaged across pixels so that the signal has only one dimension in space and one dimension in time.

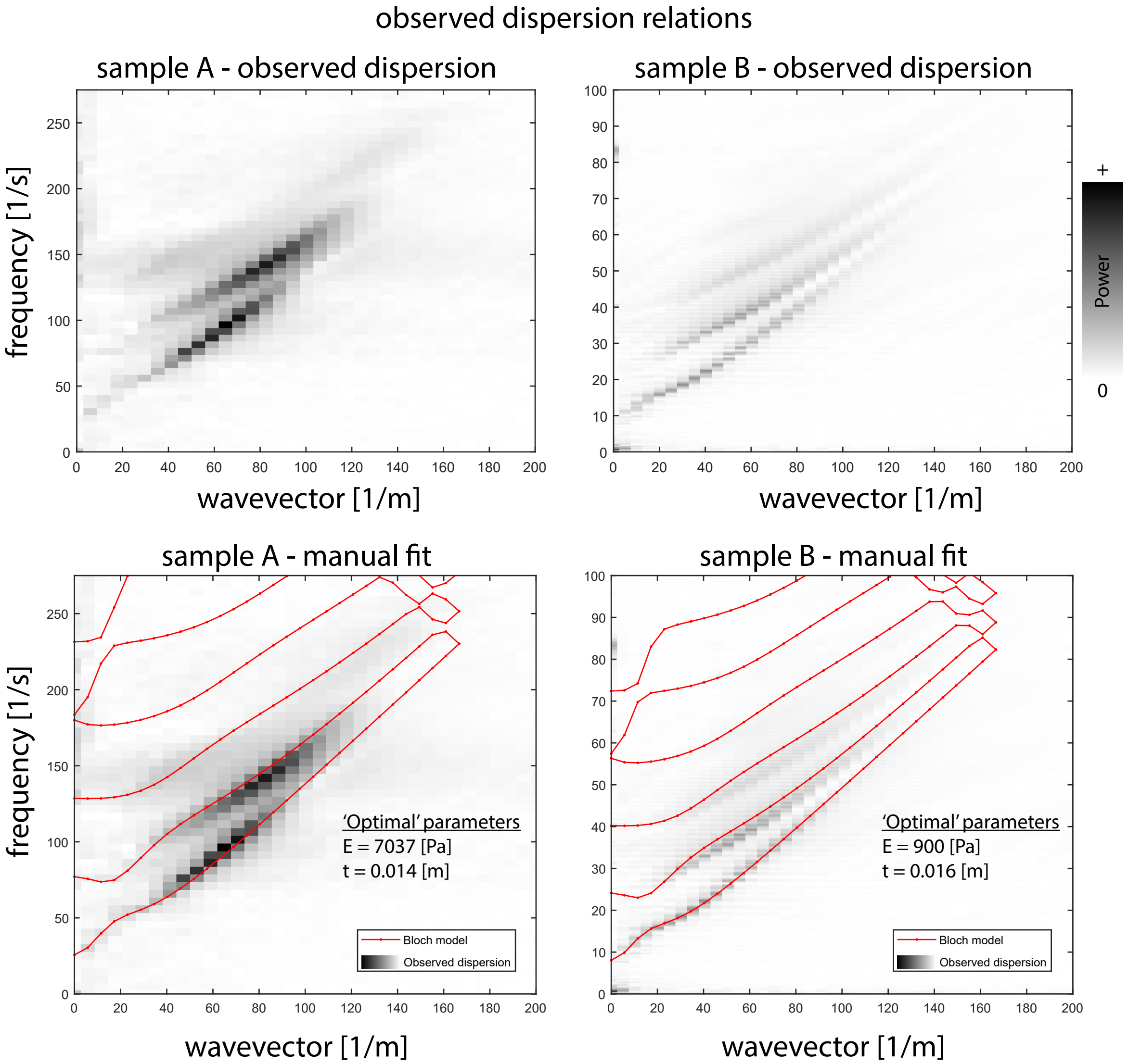


Figure 6.12: First row: two examples of experimentally observed dispersion relations. Second row: Bloch model overlayed on observations using strata parameters from a manually performed fit of the observed data.

6.6 Conclusion

Overall, this research demonstrates the feasibility of using visual data of surface waves to make inferences about subsurface features. As the availability, affordability, and quality of consumer-grade sensors (smartwatches, smartphones) increases, the massive amount of available biometric data needs scientifically grounded interpretations in order to be useful. Visual Surface Wave Tomography, even in it's preliminary stages, shows considerable promise in a future of delivering physics-based, medically relevant information for early detection via frequent at-home screening.

While the numerics of this research already show excellent results, there is room for improvement with the real-world experimental results. Future directions for this research include:

1. Improving the observation processing pipeline to better account for real-world effects, like the fact that the pixel-to-distance ratio is not constant across the frames of the video.
2. Allowing for scenarios with a spatial variation in strata properties. This can likely be achieved with *short-time fourier transform* applied to the spatial dimension of the data.
3. Incorporating three-dimensional physics into the inference problem. Extending the inference paradigm to encompass three-dimensional modes could provide important information to make this method more robust and accurate, and may allow us to estimate anisotropic qualities of layered systems.
4. Making the fitting process more computationally efficient by using surrogate models (e.g., *GPR-dispersion* surrogate model from Ch. 4) and/or implementing a fitting routine more elegant than comprehensive grid search.
5. Developing more accurate models for biological and/or hydrogel-based materials (for example, by including more complex constitutive behaviors in the inference process, like viscoelasticity, poroelasticity, anisotropy, etc).
6. Incorporate basic observational data from on-surface measurements (representing accelerometer data from a smartwatch, for example).
7. Performing experimental validations with truly affordable cameras.
8. Using numerical experiments to determine if a larger number of strata can be characterized, and understand what the limits surrounding this might be .

Chapter 7

CONCLUSION

7.1 Summary of contributions

This thesis has explored the integration of state-of-the-art computational methods to elastodynamic applications, from architected material modeling and design to the inference and imaging of systems based on their elastodynamic behaviors.

The application of machine learning and data-driven methods to the surrogate modeling and design of architected materials was explored. The current state of the field was assessed, and perspectives were given on the challenges associated with the design of architected materials and with the application of data-driven methodologies to this design problem. Within these perspectives, certain desirable traits for surrogate models were proposed offering a set of guiding principles for the future development of surrogate models for physical systems. Interpretable machine learning methods were developed, including shape-frequency features, unit cell templates, and hierarchical unit cell templates. These models allow for high levels of geometric interpretability, offering an interesting alternative approach in a scientific landscape mostly dominated by uninterpretable, black-box deep learning models. Next, a surrogate model, GPR-dispersion, was developed to more efficiently compute the dispersion relations of engineered material systems, aimed at the application of optimization-based design of multi-functional elastodynamic engineered materials. GPR-dispersion was constructed to exhibit several of the desirable traits for surrogate models mentioned prior, including representation invariance, incorporation of physics-based modeling, data efficiency, interpretability, and the provision of uncertainty metrics and gradients for added utility when used for inverse design. It was tested against traditional purely-physics based methods, and another existing surrogate model based on deep learning.

Then, with a change of theme, two methods were introduced to characterize unseen interiors of material systems via the partial visual observation of their surface dynamics through monocular video. Visual Vibration Tomography (VVT) offers a methodology for estimating the relative elastic modulus and density through the volume of an object by partially observing surface displacements through monocular video, estimating the vibrational modes of the object, and fitting a physics-based model to the observations. The performance of VVT was demonstrated extensively in simulated experiments and real-world experiments. Inspired by VVT, Visual Surface Wave Tomography (VSWT) offers a methodology for inferring the subsurface material properties of layered material systems, with the application of inexpensive health tracking in mind. VSWT partially observes the surface wave dynamics of the system through monocular video to build an understanding of the medium's dispersive properties. Subsequently, a physics-based model is fit to the observed data to yield estimations of the layer moduli and thicknesses. The performance, sensitivity, and limits of VSWT were explored thoroughly in simulated experiments, and real-world proof-of-concept experiments were demonstrated with gelatin-based hydrogel phantom materials, emulating the material properites of biological soft tissue.

7.2 Perspectives on future research directions for surrogate modeling and material design

With the ever-accelerating development of data-driven methods, many new possibilites for the modeling and design of engineering are opened. As discussed in this thesis, data-driven surrogate modeling is a critical facet of this effort. This section discusses perspectives on new possibilities for research in the surrogate modeling of physical systems.

7.2.1 Intersection and intercompatibility of advanced methods for material design

While impressive advancements have been made on many fronts of material design, there are gaps between these fronts. On one front, researchers have delivered impressive advancements in areas related to the traditional methods of material design. These areas include principled physics-based modeling [137–141], and implementation of these methods in extremely optimized and massively parallel computing environments [83]. On a separate front, researchers have developed increasingly advanced data-driven surrogate models both for the efficient forward modeling and inverse design of architected materials [84, 94, 154, 232]. As of yet, these two fronts have not fully intersected. As discussed in this thesis, the advanced methods developed in each front often lack intercompatibility with each other. Instead of working together with each other to achieve compounding positive effects, they are often constructed in a paradigm that ignores the other camp, except to compete. An extremely compelling paradigm for future work is the engineering of intercompatibility and intersection between these two fronts to form hybridized design systems that benefit in a combined way from the robust physical basis of traditional methods, the massively parallel implementations of state-of-the-art design optimization routines, and the evident power of data-driven learning methods.

7.2.2 Representation invariant surrogate models

As discussed in this thesis, an overwhelming number of surrogate models for physical systems rely on specific discretization schemes, or specific parametric representation schemes that contrain the model's use cases to a narrow slice of design space. These reliances harm the practical applicability of surrogate models, and their ability to be reused for new applications. Especially in the context of data-driven approaches, it is extremely desirable for a model, which may have cost a large amount of data to construct, to be usable in a wide variety of settings. Models that are cornered into their own narrow region of design space, while academically interesting, are often not useful in practice.

Continued investment in representation invariant models is a promising paradigm to stick to for the future development of surrogate models for physical systems. Representation invariance can be built directly into the architecture of the model, as in the case of operator learning, or can be implicitly embedded by incorporating data-driven models "on top" of physical models, as in the case of GPR-dispersion. The discovery of new mechanisms by which to introduce representation invariance would be highly beneficial to the design of architected materials.

7.2.3 Incorporation of empirical physical data into surrogate models

This thesis explored the importance of incorporating physical laws into data-driven surrogate models. With all of the excitement surrounding data-driven methods, it is critical that we keep them grounded to principled and interpretable rules. However, there are plenty of physical that are not currently well understood. In these cases, surrogate models can be trained on empirical data in order to represent physics that is not well-represented by current physics-based models. A major challenge scientists face when attempting this is that empirical data can be extremely expensive, and data-driven methods often require lots of data to train adequately. It may be interesting to study how transfer learning can be employed to alleviate this, training a model initially on a large volume of synthetic data with reasonably faithful physics, then fine tuning the resulting model on a small volume of empirical data. This could open interesting avenues for research in the context of architected material design to leverage more advanced physical phenomena, and could even spill over to have implications for elastodynamic imaging and inference problems (as discussed in the following section, sec. 7.3.2)

7.3 Perspectives on future research directions for inexpensive material characterization

VVT and VSWT propose compelling visual approaches to the mechanical characterization of systems using elastodynamic information. They also open the doors to a variety of new avenues for future research. This section explores some of the most promising directions.

7.3.1 Incorporation of surrogate models for inference problems

The formulations of the inference problems utilized by these novel techniques rely on the evaluation of physical models in order to fit a physics-based hypothesis to explain the observed data, and yield an inference. The practical implementation of these methods usually rely on optimization formulations, which in turn demand the physical models to be evaluated repeatedly. Even when implemented in highly optimized ways (as in this thesis), this can lead to significant computational cost, which may discourage the real-world deployment of the method. An interesting future avenue could leverage data-driven surrogate models to make this process more efficient and compelling in a low-powered computing settings, such as at-home health monitoring.

7.3.2 Development of higher fidelity physical models for inference problems

As stated, inference problems rely on the idea that a physics-based hypothesis can be fit to a set of observed data. For example, in the research presented in this thesis, the physics-based hypothesis were rooted in linear-elasticity. Gaps between real-world physics and linear-elasticity were partially responsible decreases in the real-world performance of the inference methods, as suggested by the delta between the performance of the methods in simulated settings versus real settings. Not only would the incorporation of more accurate physical models likely lead to more robust inference performance, but would also allow the characterization of more complex materials. For example, each of these methods could be extended to infer the viscoelastic characteristics of systems by estimating the parameters of basic damping models, or could relax the assumption of isotropy to infer directional characteristics of systems.

With these extensions would come challenges. Selection of a physical model appropriate to a specific system is a non-trivial task, and may (at least initially) require some amount of intuitive selection. Even with an appropriate physical model selected, deviation from simple models like linear-elasticity often means that the parameter fitting process will have a vastly larger parameter space to search through, which could lead to practical computational challenges. It's possible that the development of data-driven models based on empirical observations, as desicribed in sec. 7.2.3 could be useful here.

7.3.3 Real-world extension to more complex geometries

VVT and VSWT both leverage strong geometric priors in order to function. VVT assumes a geometry is known, and results were presented demonstrating the error induced when the geometric prior is incorrect. Real-world experiments were demonstrated on circular membrane systems and cubic solid systems with known geometries, and simple occlusion relationships between the surfaces. It would be interesting to combine this method with geometry estimation technologies (for example, LIDAR-based) that could provide and estimation of this geometric prior to form a more complete imaging system. VSWT assumes a very simple class of geometry, layered material systems with layer thicknesses that are constant in the in-plane directions. A future extension of this method could allow variability in layer thicknesses, for example, through the use of Short-Time Fourier Transform methods (applied in the spatial dimension) to understand the spatial variations in the spectral representations of surface wave dynamics. It also assumes that the elastodynamics of the system are well-represented by quasi-two-dimensional approximations like plane-strain, which could be allieviated in the future by the use of three-dimensional Bloch models.

7.3.4 Combined inference with multiple types of observed data

While it is impressive that VVT and VSWT can make inferences and feature detections with only monocular video taken from one angle, and it would be interesting to see the full limits of how far visual information can take us, it would also be impactful to develop methods to allow the combined use of additional observations of the system in question. This could be as simple as viewing the system from an additional angle (thereby gathering a more complete understanding of the surface data), or allowing the use of multi-modal data from one or more inexpensive sensors placed on the surface of the system. For example, in the application of health tracking via wearables, VSWT may benefit from the addition of basic, low-frequency accelerometer data that may be available from a smartwatch. Incorporating these additional observations could be implemented through a multi-faceted optimization approach, fitting different physical models to each type of observation, as appropriate, and could provide each method with more information to yield a higher quality inference.

BIBLIOGRAPHY

[1] D.C. Malocha. “SAW/BAW Acoustoelectronic Technology for Filters and Communication Systems”. In: *2010 IEEE 11th Annual Wireless and Microwave Technology Conference (WAMICON)*. 2010 IEEE 11th Annual Wireless and Microwave Technology Conference (WAMICON). Apr. 2010, pp. 1–7. DOI: `10.1109/WAMICON.2010.5461848`. URL: `https://doi.org/10.1109/WAMICON.2010.5461848` (visited on 04/25/2024).

[2] David Morgan. *Surface Acoustic Wave Filters: With Applications to Electronic Communications and Signal Processing*. Academic Press, July 27, 2010. 448 pp. ISBN: 978-0-08-055013-8. Google Books: `ITLn6tj4SDQC`.

[3] Colin Campbell. *Surface Acoustic Wave Devices and Their Signal Processing Applications*. Elsevier, Dec. 2, 2012. 485 pp. ISBN: 978-0-323-14866-5.

[4] Keiiti Aki and Paul G. Richards. “Quantitative Seismology, 2nd Ed.” In: *Quantitative Seismology* (2002). URL: `https://ui.adsabs.harvard.edu/abs/2002quse.book.....A/abstract` (visited on 04/25/2024).

[5] Gareth J. Crutchley and Heidrun Kopp. “Reflection and Refraction Seismic Methods”. In: *Submarine Geomorphology*. Ed. by Aaron Micallef, Sebastian Krastel, and Alessandra Savini. Cham: Springer International Publishing, 2018, pp. 43–62. ISBN: 978-3-319-57852-1. DOI: `10.1007/978-3-319-57852-1_4`. URL: `https://doi.org/10.1007/978-3-319-57852-1_4` (visited on 04/19/2024).

[6] Choon Park. “MASW for Geotechnical Site Investigation”. In: *The Leading Edge* 32.6 (June 1, 2013), pp. 656–662. ISSN: 1070-485X. DOI: `10.1190/tle32060656.1`. URL: `https://doi.org/10.1190/tle32060656.1` (visited on 04/19/2024).

[7] Nikolai M. Shapiro et al. “High-Resolution Surface-Wave Tomography from Ambient Seismic Noise”. In: *Science* 307.5715 (Mar. 11, 2005), pp. 1615–1618. DOI: `10.1126/science.1108339`. URL: `https://www.science.org/doi/full/10.1126/science.1108339` (visited on 04/19/2024).

[8] Soheil Nazarian and Kenneth Stokoe. “Evaluation of Moduli and Thicknesses of Pavement Systems by Spectral-Analysis-of-Surface-Waves Method”. In: *The Study of New Technologies for Pavement Evaluation Research Project 3-8-80-256, as conducted for Texas State Department of Highways and Public Transportation, by the CENTER FOR TRANSPORTATION RESEARCH BUREAU OF ENGINEERING RESEARCH THE UNIVERSITY OF TEXAS AT AUSTIN* (1983).

[9] William Thomson, Marie Dahleh, and C. Padmanabhan. *Theory of Vibration with Applications*. 5th ed. Prentice Hall.

[10] Vincent Wai-Sun Wong and Henry Lik-Yuen Chan. “Transient Elastography”. In: *Journal of Gastroenterology and Hepatology* 25.11 (2010), pp. 1726–1731. ISSN: 1440-1746. DOI: `10.1111/j.1440-1746.2010.06437.x`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1440-1746.2010.06437.x` (visited on 04/19/2024).

[11] Mihra S. Taljanovic et al. "Shear-Wave Elastography: Basic Physics and Musculoskeletal Applications". In: *RadioGraphics* 37.3 (May 2017), pp. 855–870. ISSN: 0271-5333. DOI: `10.1148/rg.2017160116`. URL: `https://pubs.rsna.org/doi/abs/10.1148/rg.2017160116` (visited on 04/19/2024).

[12] Yogesh K. Mariappan, Kevin J. Glaser, and Richard L. Ehman. "Magnetic Resonance Elastography: A Review". In: *Clinical Anatomy* 23.5 (2010), pp. 497–511. ISSN: 1098-2353. DOI: `10.1002/ca.21006`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/ca.21006` (visited on 04/19/2024).

[13] Christian Chaussy and Egbert Schmiedt. "Extracorporeal Shock Wave Lithotripsy (ESWL) for Kidney Stones. An Alternative to Surgery?" In: *Urologic radiology* 6.1 (Dec. 1, 1984), pp. 80–87. ISSN: 0171-1091. DOI: `10.1007/BF02923707`. URL: `https://doi.org/10.1007/BF02923707` (visited on 04/25/2024).

[14] M. Miniaci et al. "Valley-based splitting of topologically protected helical waves in elastic plates". en. In: *Physical Review B* 100.2 (July 2019), p. 024304. ISSN: 2469-9950, 2469-9969. DOI: `10.1103/PhysRevB.100.024304`. URL: `https://link.aps.org/doi/10.1103/PhysRevB.100.024304` (visited on 05/02/2020).

[15] Gunho Kim et al. "Poroelastic microlattices for underwater wave focusing". en. In: *Extreme Mechanics Letters* 49 (Nov. 2021), p. 101499. ISSN: 2352-4316. DOI: `10.1016/j.eml.2021.101499`. URL: `https://www.sciencedirect.com/science/article/pii/S2352431621001930` (visited on 03/20/2023).

[16] Erik Andreassen et al. "Optimization of directional elastic energy propagation". en. In: *Journal of Sound and Vibration* 379 (Sept. 2016), pp. 53–70. ISSN: 0022-460X. DOI: `10.1016/j.jsv.2016.03.002`. URL: `https://www.sciencedirect.com/science/article/pii/S0022460X16002315` (visited on 09/13/2021).

[17] Sai Sharan Injeti et al. "Tuning acoustic impedance in load-bearing structures". In: *arXiv: [cond-mat, physics:physics]* (June 2021). arXiv: 2106.10573. URL: `http://arxiv.org/abs/2106.10573` (visited on 04/07/2022).

[18] Tian Xu et al. "Ultrasonic Stimulation of the Brain to Enhance the Release of Dopamine – A Potential Novel Treatment for Parkinson's Disease". In: *Ultrasonics Sonochemistry* 63 (May 1, 2020), p. 104955. ISSN: 1350-4177. DOI: `10.1016/j.ultsonch.2019.104955`. URL: `https://doi.org/10.1016/j.ultsonch.2019.104955` (visited on 04/25/2024).

[19] Hui Zhou et al. "Noninvasive Ultrasound Deep Brain Stimulation for the Treatment of Parkinson's Disease Model Mouse". In: *Research* 2019 (July 9, 2019). DOI: `10.34133/2019/1748489`. URL: `https://spj.science.org/doi/full/10.34133/2019/1748489` (visited on 04/25/2024).

[20] Guglielmo Foffani et al. "Focused Ultrasound in Parkinson's Disease: A Twofold Path toward Disease Modification". In: *Movement Disorders* 34.9 (2019), pp. 1262–1273. ISSN: 1531-8257. DOI: `10.1002/mds.27805`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/mds.27805` (visited on 04/25/2024).

[21] Pawel Piotr Dobrakowski et al. “MR-Guided Focused Ultrasound: A New Generation Treatment of Parkinson’s Disease, Essential Tremor and Neuropathic Pain”. In: *Interventional Neuroradiology* 20.3 (May 1, 2014), pp. 275–282. ISSN: 1591-0199. DOI: `10.15274/INR-2014-10033`. URL: `https://doi.org/10.15274/INR-2014-10033` (visited on 04/25/2024).

[22] Lauren J. Delaney et al. “Making Waves: How Ultrasound-Targeted Drug Delivery Is Changing Pharmaceutical Approaches”. In: *Materials Advances* 3.7 (2022), pp. 3023–3040. DOI: `10.1039/D1MA01197A`. URL: `https://pubs.rsc.org/en/content/articlelanding/2022/ma/d1ma01197a` (visited on 04/25/2024).

[23] Riaz Ahmed, Fariha Mir, and Sourav Banerjee. “A Review on Energy Harvesting Approaches for Renewable Energies from Ambient Vibrations and Acoustic Waves Using Piezoelectricity”. In: *Smart Materials and Structures* 26.8 (July 2017), p. 085031. ISSN: 0964-1726. DOI: `10.1088/1361-665X/aa7bfb`. URL: `https://dx.doi.org/10.1088/1361-665X/aa7bfb` (visited on 04/25/2024).

[24] X. Shi, M. Zhang, and L. F. Wei. “Quantum Computation with Moving Quantum Dots Generated by Surface Acoustic Waves”. In: *Physical Review A* 84.6 (Dec. 15, 2011), p. 062310. DOI: `10.1103/PhysRevA.84.062310`. URL: `https://link.aps.org/doi/10.1103/PhysRevA.84.062310` (visited on 04/25/2024).

[25] P. A. Deymier et al. “Setting the Stage for Materials Simulation Using Acoustic Metamaterials Digital Quantum Analogue Computing Platforms”. In: *Modelling and Simulation in Materials Science and Engineering* 30.8 (Oct. 2022), p. 084003. ISSN: 0965-0393. DOI: `10.1088/1361-651X/ac991e`. URL: `https://dx.doi.org/10.1088/1361-651X/ac991e` (visited on 04/25/2024).

[26] J. B. Pendry. “Negative Refraction Makes a Perfect Lens”. In: *Physical Review Letters* 85.18 (Oct. 30, 2000), pp. 3966–3969. DOI: `10.1103/PhysRevLett.85.3966`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.85.3966` (visited on 02/04/2024).

[27] Gareth Jones and Emma C. Teeling. “The Evolution of Echolocation in Bats”. In: *Trends in Ecology & Evolution* 21.3 (Mar. 1, 2006), pp. 149–156. ISSN: 0169-5347. DOI: `10.1016/j.tree.2006.01.001`. pmid: `16701491`. URL: `https://www.cell.com/trends/ecology-evolution/abstract/S0169-5347(06)00021-8` (visited on 04/25/2024).

[28] Whitlow W. L. Au. “Echolocation in Dolphins”. In: *Hearing by Whales and Dolphins*. Ed. by Whitlow W. L. Au, Richard R. Fay, and Arthur N. Popper. New York, NY: Springer, 2000, pp. 364–408. ISBN: 978-1-4612-1150-1. DOI: `10.1007/978-1-4612-1150-1_9`. URL: `https://doi.org/10.1007/978-1-4612-1150-1_9` (visited on 04/25/2024).

[29] P. T. Madsen et al. “Echolocation in Blainville’s Beaked Whales (Mesoplodon Densirostris)”. In: *Journal of Comparative Physiology A* 199.6 (June 1, 2013), pp. 451–469. ISSN: 1432-1351. DOI: `10.1007/s00359-013-0824-8`. URL: `https://doi.org/10.1007/s00359-013-0824-8` (visited on 04/25/2024).

[30] Chung-Huey Wu and Damian O. Elias. “Vibratory Noise in Anthropogenic Habitats and Its Effect on Prey Detection in a Web-Building Spider”. In: *Animal Behaviour* 90 (Apr. 1, 2014), pp. 47–56. ISSN: 0003-3472. DOI: `10.1016/j.anbehav.2014.01.006`. URL:

`https://www.sciencedirect.com/science/article/pii/S0003347214000402` (visited on 04/25/2024).

[31] Rainer Foelix. *Biology of Spiders*. Oxford University Press, USA, May 5, 2011. 428 pp. ISBN: 978-0-19-973482-5. Google Books: `eOUVDAAAQBAJ`.

[32] Caitlin O'Connell-Rodwell, Xiying Guan, and Sunil Puria. "Vibrational Communication in Elephants: A Case for Bone Conduction". In: *Biotremology: Studying Vibrational Behavior*. Ed. by Peggy S. M. Hill et al. Cham: Springer International Publishing, 2019, pp. 259–276. ISBN: 978-3-030-22293-2. DOI: `10.1007/978-3-030-22293-2_13`. URL: `https://doi.org/10.1007/978-3-030-22293-2_13` (visited on 04/25/2024).

[33] Caitlin E. O'Connell-Rodwell et al. "Wild African Elephants (Loxodonta Africana) Discriminate between Familiar and Unfamiliar Conspecific Seismic Alarm Calls". In: *The Journal of the Acoustical Society of America* 122.2 (Aug. 1, 2007), pp. 823–830. ISSN: 0001-4966. DOI: `10.1121/1.2747161`. URL: `https://doi.org/10.1121/1.2747161` (visited on 04/25/2024).

[34] Beth Mortimer et al. "Noise Matters: Elephants Show Risk-Avoidance Behaviour in Response to Human-Generated Seismic Cues". In: *Proceedings of the Royal Society B: Biological Sciences* 288.1953 (June 30, 2021), p. 20210774. DOI: `10.1098/rspb.2021.0774`. URL: `https://royalsocietypublishing.org/doi/full/10.1098/rspb.2021.0774` (visited on 04/25/2024).

[35] Tom Reuter, Sirpa Nummela, and Simo Hemilä. "Elephant Hearing". In: *The Journal of the Acoustical Society of America* 104.2 (Aug. 1, 1998), pp. 1122–1123. ISSN: 0001-4966. DOI: `10.1121/1.423341`. URL: `https://doi.org/10.1121/1.423341` (visited on 04/25/2024).

[36] Carl J. Erickson et al. "Percussive Foraging: Stimuli for Prey Location by Aye-Ayes (Daubentonia Madagascariensis)". In: *International Journal of Primatology* 19.1 (Feb. 1, 1998), pp. 111–122. ISSN: 1573-8604. DOI: `10.1023/A:1020363128240`. URL: `https://doi.org/10.1023/A:1020363128240` (visited on 04/25/2024).

[37] Don Dissanayake. *Acoustic Waves*. BoD – Books on Demand, Sept. 28, 2010. 480 pp. ISBN: 978-953-307-111-4. Google Books: `NuqODwAAQBAJ`.

[38] Bernard Masserey, Christian Raemy, and Paul Fromme. "High-Frequency Guided Ultrasonic Waves for Hidden Defect Detection in Multi-Layered Aircraft Structures". In: *Ultrasonics* 54.7 (Sept. 1, 2014), pp. 1720–1728. ISSN: 0041-624X. DOI: `10.1016/j.ultras.2014.04.023`. URL: `https://www.sciencedirect.com/science/article/pii/S0041624X14001140` (visited on 04/25/2024).

[39] Glenn J. Rix et al. *Manual on Subsurface Investigations*. Washington, D.C.: Transportation Research Board, Feb. 6, 2019, p. 25379. ISBN: 978-0-309-48947-8. DOI: `10.17226/25379`. URL: `https://www.nap.edu/catalog/25379` (visited on 04/15/2024).

[40] Kenneth Hoyt et al. "Tissue Elasticity Properties as Biomarkers for Prostate Cancer". In: *Cancer Biomarkers* 4.4-5 (Jan. 1, 2008), pp. 213–225. ISSN: 1574-0153. DOI: `10.3233/CBM-2008-44-505`. URL: `https://content.iospress.com/articles/cancer-biomarkers/dma-a538` (visited on 04/19/2024).

[41] Sangwoo Kwon et al. “Comparison of Cancer Cell Elasticity by Cell Type”. In: *Journal of Cancer* 11.18 (July 11, 2020), pp. 5403–5412. ISSN: 1837-9664. DOI: `10.7150/jca.45897`. pmid: `32742487`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7391204/` (visited on 04/19/2024).

[42] Penelope C. Georges et al. “Increased Stiffness of the Rat Liver Precedes Matrix Deposition: Implications for Fibrosis”. In: *American Journal of Physiology-Gastrointestinal and Liver Physiology* 293.6 (Dec. 2007), G1147–G1154. ISSN: 0193-1857. DOI: `10.1152/ajpgi.00032.2007`. URL: `https://journals.physiology.org/doi/full/10.1152/ajpgi.00032.2007` (visited on 04/19/2024).

[43] Enis Kostallari et al. “Stiffness Is Associated with Hepatic Stellate Cell Heterogeneity during Liver Fibrosis”. In: *American Journal of Physiology-Gastrointestinal and Liver Physiology* 322.2 (Feb. 2022), G234–G246. ISSN: 0193-1857. DOI: `10.1152/ajpgi.00254.2021`. URL: `https://journals.physiology.org/doi/full/10.1152/ajpgi.00254.2021` (visited on 04/19/2024).

[44] Shan Sun et al. “Biomechanics and Functionality of Hepatocytes in Liver Cirrhosis”. In: *Journal of Biomechanics*. Functional Tissue Engineering 47.9 (June 27, 2014), pp. 2205–2210. ISSN: 0021-9290. DOI: `10.1016/j.jbiomech.2013.10.050`. URL: `https://www.sciencedirect.com/science/article/pii/S0021929013005423` (visited on 04/19/2024).

[45] Chandak Upagupta et al. “Matrix Abnormalities in Pulmonary Fibrosis”. In: *European Respiratory Review* 27.148 (June 30, 2018). ISSN: 0905-9180, 1600-0617. DOI: `10.1183/16000617.0033-2018`. pmid: `29950306`. URL: `https://err.ersjournals.com/content/27/148/180033` (visited on 04/19/2024).

[46] Jade Jaffar et al. “Greater Cellular Stiffness in Fibroblasts from Patients with Idiopathic Pulmonary Fibrosis”. In: *American Journal of Physiology-Lung Cellular and Molecular Physiology* 315.1 (July 2018), pp. L59–L65. ISSN: 1040-0605. DOI: `10.1152/ajplung.00030.2018`. URL: `https://journals.physiology.org/doi/full/10.1152/ajplung.00030.2018` (visited on 04/19/2024).

[47] Richard S. Nho et al. “Biomechanical Force and Cellular Stiffness in Lung Fibrosis”. In: *The American Journal of Pathology* 192.5 (May 1, 2022), pp. 750–761. ISSN: 0002-9440. DOI: `10.1016/j.ajpath.2022.02.001`. URL: `https://www.sciencedirect.com/science/article/pii/S0002944022000505` (visited on 04/19/2024).

[48] Katelyn Hasse et al. “Feasibility of Deriving a Novel Imaging Biomarker Based on Patient-Specific Lung Elasticity for Characterizing the Degree of COPD in Lung SBRT Patients”. In: *British Journal of Radiology* 92.1094 (Feb. 1, 2019), p. 20180296. ISSN: 0007-1285. DOI: `10.1259/bjr.20180296`. URL: `https://doi.org/10.1259/bjr.20180296` (visited on 04/19/2024).

[49] P. N. Black et al. “Changes in Elastic Fibres in the Small Airways and Alveoli in COPD”. In: *European Respiratory Journal* 31.5 (May 1, 2008), pp. 998–1004. ISSN: 0903-1936, 1399-3003. DOI: `10.1183/09031936.00017207`. pmid: `18216063`. URL: `https://erj.ersjournals.com/content/31/5/998` (visited on 04/19/2024).

[50] Michel E. Safar. “Arterial Stiffness as a Risk Factor for Clinical Hypertension”. In: *Nature Reviews Cardiology* 15.2 (Feb. 2018), pp. 97–105. ISSN: 1759-5010. DOI: `10.1038/nrcardio.2017.155`. URL: `https://www.nature.com/articles/nrcardio.2017.155` (visited on 04/19/2024).

[51] S. -I. Murtada et al. “Differential Biomechanical Responses of Elastic and Muscular Arteries to Angiotensin II-induced Hypertension”. In: *Journal of Biomechanics* 119 (Apr. 15, 2021), p. 110297. ISSN: 0021-9290. DOI: `10.1016/j.jbiomech.2021.110297`. URL: `https://www.sciencedirect.com/science/article/pii/S0021929021000774` (visited on 04/19/2024).

[52] Pierre Boutouyrie et al. “Arterial Stiffness and Cardiovascular Risk in Hypertension”. In: *Circulation Research* 128.7 (Apr. 2, 2021), pp. 864–886. DOI: `10.1161/CIRCRESAHA.121.318061`. URL: `https://doi.org/10.1161/CIRCRESAHA.121.318061` (visited on 04/19/2024).

[53] Carlo Palombo and Michaela Kozakova. “Arterial Stiffness, Atherosclerosis and Cardiovascular Risk: Pathophysiologic Mechanisms and Emerging Clinical Indications”. In: *Vascular Pharmacology* 77 (Feb. 1, 2016), pp. 1–7. ISSN: 1537-1891. DOI: `10.1016/j.vph.2015.11.083`. URL: `https://www.sciencedirect.com/science/article/pii/S1537189115003328` (visited on 04/19/2024).

[54] Daniel A. Duprez and Jay N. Cohn. “Arterial Stiffness as a Risk Factor for Coronary Atherosclerosis”. In: *Current Atherosclerosis Reports* 9.2 (Apr. 1, 2007), pp. 139–144. ISSN: 1534-6242. DOI: `10.1007/s11883-007-0010-y`. URL: `https://doi.org/10.1007/s11883-007-0010-y` (visited on 04/19/2024).

[55] N. H. Varady and A. J. Grodzinsky. “Osteoarthritis Year in Review 2015: Mechanics”. In: *Osteoarthritis and Cartilage* 24.1 (Jan. 1, 2016), pp. 27–35. ISSN: 1063-4584. DOI: `10.1016/j.joca.2015.08.018`. URL: `https://www.sciencedirect.com/science/article/pii/S1063458415013175` (visited on 04/19/2024).

[56] Lilian Lacourpaille et al. “Non-Invasive Assessment of Muscle Stiffness in Patients with Duchenne Muscular Dystrophy”. In: *Muscle & Nerve* 51.2 (2015), pp. 284–286. ISSN: 1097-4598. DOI: `10.1002/mus.24445`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/mus.24445` (visited on 04/19/2024).

[57] Kelley M. Virgilio et al. “Multiscale Models of Skeletal Muscle Reveal the Complex Effects of Muscular Dystrophy on Tissue Mechanics and Damage Susceptibility”. In: *Interface Focus* 5.2 (Apr. 6, 2015), p. 20140080. DOI: `10.1098/rsfs.2014.0080`. URL: `https://royalsocietypublishing.org/doi/full/10.1098/rsfs.2014.0080` (visited on 04/19/2024).

[58] Mohammed Al-khafajiy et al. “Remote Health Monitoring of Elderly through Wearable Sensors”. In: *Multimedia Tools and Applications* 78.17 (Sept. 1, 2019), pp. 24681–24706. ISSN: 1573-7721. DOI: `10.1007/s11042-018-7134-7`. URL: `https://doi.org/10.1007/s11042-018-7134-7` (visited on 04/26/2024).

[59] Alexandros Pantelopoulos and Nikolaos G. Bourbakis. “A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 40.1 (Jan. 2010), pp. 1–12. ISSN: 1558-2442. DOI: `10.1109/TSMCC.2009.2032660`. URL: `https://doi.org/10.1109/TSMCC.2009.2032660` (visited on 04/26/2024).

[60] Yue Liao et al. “The Future of Wearable Technologies and Remote Monitoring in Health Care”. In: *American Society of Clinical Oncology Educational Book* 39 (May 2019), pp. 115–121. ISSN: 1548-8748. DOI: `10.1200/EDBK_238919`. URL: `https://ascopubs.org/doi/full/10.1200/EDBK_238919` (visited on 04/26/2024).

[61] Jessilyn Dunn, Ryan Runge, and Michael Snyder. “Wearables and the Medical Revolution”. In: *Personalized Medicine* 15.5 (Sept. 2018), pp. 429–448. ISSN: 1741-0541. DOI: `10.2217/pme-2018-0044`. URL: `https://www.futuremedicine.com/doi/full/10.2217/pme-2018-0044` (visited on 04/26/2024).

[62] David Olczuk and Ronny Priefer. “A History of Continuous Glucose Monitors (CGMs) in Self-Monitoring of Diabetes Mellitus”. In: *Diabetes & Metabolic Syndrome: Clinical Research & Reviews* 12.2 (Apr. 1, 2018), pp. 181–187. ISSN: 1871-4021. DOI: `10.1016/j.dsx.2017.09.005`. URL: `https://www.sciencedirect.com/science/article/pii/S187140211730303X` (visited on 04/26/2024).

[63] Brooke H. McAdams and Ali A. Rizvi. “An Overview of Insulin Pumps and Glucose Sensors for the Generalist”. In: *Journal of Clinical Medicine* 5.1 (1 Jan. 2016), p. 5. ISSN: 2077-0383. DOI: `10.3390/jcm5010005`. URL: `https://www.mdpi.com/2077-0383/5/1/5` (visited on 04/26/2024).

[64] Johnson Francis and Sven Reek. “Wearable Cardioverter Defibrillator: A Life Vest till the Life Boat (ICD) Arrives”. In: *Indian Heart Journal* 66.1 (Jan. 1, 2014), pp. 68–72. ISSN: 0019-4832. DOI: `10.1016/j.ihj.2013.12.050`. URL: `https://www.sciencedirect.com/science/article/pii/S0019483213004471` (visited on 04/26/2024).

[65] *Oura Ring. Smart Ring for Fitness, Stress, Sleep & Health.* Oura Ring. URL: `https://ouraring.com` (visited on 04/26/2024).

[66] *Natural Cycles: Natural Birth Control | No Hormones or Side Effects*. Natural Cycles. Mar. 23, 2020. URL: `https://www.naturalcycles.com/` (visited on 04/26/2024).

[67] Jack T. Pearson et al. “Contraceptive Effectiveness of an FDA-Cleared Birth Control App: Results from the Natural Cycles U.S. Cohort”. In: *Journal of Women’s Health* 30.6 (June 2021), pp. 782–788. ISSN: 1540-9996. DOI: `10.1089/jwh.2020.8547`. URL: `https://www.liebertpub.com/doi/abs/10.1089/jwh.2020.8547` (visited on 04/26/2024).

[68] Jack T. Pearson et al. “Natural Cycles App: Contraceptive Outcomes and Demographic Analysis of UK Users”. In: *The European Journal of Contraception & Reproductive Health Care* 26.2 (Mar. 4, 2021), pp. 105–110. ISSN: 1362-5187. DOI: `10.1080/13625187.2020.1867844`. pmid: `33539252`. URL: `https://doi.org/10.1080/13625187.2020.1867844` (visited on 04/26/2024).

[69] Hsiao-Chuan Liu et al. “Wearable Bioadhesive Ultrasound Shear Wave Elastography”. In: *Science Advances* 10.6 (Feb. 9, 2024), eadk8426. DOI: `10.1126/sciadv.adk8426`. URL: `https://www.science.org/doi/full/10.1126/sciadv.adk8426` (visited on 04/26/2024).

[70] V.G. Veselago. “The Electrodynamics of Substances with Simultaneously Negative Values of \epsilon and \mu”. In: *Soviet Physics Uspekhi* 92 (1968).

[71] D. R. Smith et al. “Composite Medium with Simultaneously Negative Permeability and Permittivity”. In: *Physical Review Letters* 84.18 (May 1, 2000), pp. 4184–4187. DOI: `10.1103/PhysRevLett.84.4184`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.84.4184` (visited on 02/04/2024).

[72] D. Schurig et al. “Metamaterial Electromagnetic Cloak at Microwave Frequencies”. In: *Science* 314.5801 (Nov. 10, 2006), pp. 977–980. DOI: `10.1126/science.1133628`. URL: `https://www.science.org/doi/10.1126/science.1133628` (visited on 02/04/2024).

[73] Roderic Lakes. “Foam Structures with a Negative Poisson’s Ratio”. In: *Science* 235.4792 (Feb. 27, 1987), pp. 1038–1040. ISSN: 0036-8075, 1095-9203. DOI: `10.1126/science.235.4792.1038`. URL: `https://www.science.org/doi/10.1126/science.235.4792.1038` (visited on 02/04/2024).

[74] Tobias Frenzel, Muamer Kadic, and Martin Wegener. “Three-Dimensional Mechanical Metamaterials with a Twist”. In: *Science* 358.6366 (Nov. 24, 2017), pp. 1072–1074. DOI: `10.1126/science.aao4640`. URL: `https://www.science.org/doi/full/10.1126/science.aao4640` (visited on 02/07/2024).

[75] S. Brûlé et al. “Experiments on Seismic Metamaterials: Molding Surface Waves”. In: *Physical Review Letters* 112.13 (Mar. 31, 2014), p. 133901. DOI: `10.1103/PhysRevLett.112.133901`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.112.133901` (visited on 02/04/2024).

[76] Alessandro Spadoni and Chiara Daraio. “Generation and Control of Sound Bullets with a Nonlinear Acoustic Lens”. In: *Proceedings of the National Academy of Sciences* 107.16 (Apr. 20, 2010), pp. 7230–7234. DOI: `10.1073/pnas.1001514107`. URL: `https://www.pnas.org/doi/abs/10.1073/pnas.1001514107` (visited on 04/29/2024).

[77] Yifan Wang et al. “Structured Fabrics with Tunable Mechanical Properties”. In: *Nature* 596.7871 (7871 Aug. 2021), pp. 238–243. ISSN: 1476-4687. DOI: `10.1038/s41586-021-03698-7`. URL: `https://www.nature.com/articles/s41586-021-03698-7` (visited on 02/07/2024).

[78] Johannes T. B. Overvelde et al. “A Three-Dimensional Actuated Origami-Inspired Transformable Metamaterial with Multiple Degrees of Freedom”. In: *Nature Communications* 7.1 (1 Mar. 11, 2016), p. 10929. ISSN: 2041-1723. DOI: `10.1038/ncomms10929`. URL: `https://www.nature.com/articles/ncomms10929` (visited on 02/04/2024).

[79] Connor McMahan et al. “Effective Continuum Models for the Buckling of Non-Periodic Architected Sheets That Display Quasi-Mechanism Behaviors”. In: *Journal of the Mechanics and Physics of Solids* 166 (Sept. 1, 2022), p. 104934. ISSN: 0022-5096. DOI: `10.1016/j.jmps.2022.104934`. URL: `https://www.sciencedirect.com/science/article/pii/S0022509622001314` (visited on 04/29/2024).

[80] Sebastien Guenneau, Claude Amra, and Denis Veynante. “Transformation Thermodynamics: Cloaking and Concentrating Heat Flux”. In: *Optics Express* 20.7 (Mar. 26, 2012), pp. 8207–8218. ISSN: 1094-4087. DOI: `10.1364/OE.20.008207`. URL: `https://opg.optica.org/oe/abstract.cfm?uri=oe-20-7-8207` (visited on 02/04/2024).

[81] Liwei Wang et al. “Deep Generative Modeling for Mechanistic-Based Learning and Design of Metamaterial Systems”. In: *Computer Methods in Applied Mechanics and Engineering* 372 (Dec. 1, 2020), p. 113377. ISSN: 0045-7825. DOI: `10.1016/j.cma.2020.113377`. URL: `https://www.sciencedirect.com/science/article/pii/S0045782520305624` (visited on 04/24/2024).

[82] Thiago Rios et al. “On the Efficiency of a Point Cloud Autoencoder as a Geometric Representation for Shape Optimization”. In: *2019 IEEE Symposium Series on Computational Intelligence (SSCI).* 2019 IEEE Symposium Series on Computational Intelligence (SSCI). Dec. 2019, pp. 791–798. DOI: `10.1109/SSCI44817.2019.9003161`. URL: `https://ieeexplore.ieee.org/abstract/document/9003161` (visited on 04/24/2024).

[83] Yan Lu et al. “3-D Phononic Crystals with Ultra-Wide Band Gaps”. In: *Scientific Reports* 7.1 (1 Feb. 24, 2017), p. 43407. ISSN: 2045-2322. DOI: `10.1038/srep43407`. URL: `https://www.nature.com/articles/srep43407` (visited on 02/12/2024).

[84] David Finol et al. “Deep convolutional neural networks for eigenvalue problems in mechanics”. en. In: *International Journal for Numerical Methods in Engineering* 118.5 (May 2019), pp. 258–275. ISSN: 0029-5981, 1097-0207. DOI: `10.1002/nme.6012`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6012` (visited on 04/24/2020).

[85] Minglei Lu et al. *Deep Neural Operator for Learning Transient Response of Interpenetrating Phase Composites Subject to Dynamic Loading*. Mar. 30, 2023. DOI: `10.48550/arXiv.2303.18055`. arXiv: `2303.18055 [cond-mat]`. URL: `http://arxiv.org/abs/2303.18055` (visited on 02/07/2024). preprint.

[86] Yannick Augenstein, Taavi Repän, and Carsten Rockstuhl. “A Neural Operator-Based Surrogate Solver for Free-Form Electromagnetic Inverse Design”. In: *ACS Photonics* 10.5 (May 17, 2023), pp. 1547–1557. ISSN: 2330-4022, 2330-4022. DOI: `10.1021/acsphotonics.3c00156`. arXiv: `2302.01934 [physics]`. URL: `http://arxiv.org/abs/2302.01934` (visited on 02/07/2024).

[87] Alexander C. Ogren et al. “Gaussian Process Regression as a Surrogate Model for the Computation of Dispersion Relations”. In: *Computer Methods in Applied Mechanics and Engineering* 420 (Feb. 15, 2024), p. 116661. ISSN: 0045-7825. DOI: `10.1016/j.cma.2023.116661`. URL: `https://doi.org/10.1016/j.cma.2023.116661` (visited on 04/14/2024).

[88] Sensong An et al. “Multifunctional Metasurface Design with a Generative Adversarial Network”. In: *Advanced Optical Materials* 9.5 (2021), p. 2001433. ISSN: 2195-1071. DOI: `10.1002/adom.202001433`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/adom.202001433` (visited on 04/29/2024).

[89] Sabrina Chin-yun Shen and Markus J. Buehler. “Nature-Inspired Architected Materials Using Unsupervised Deep Learning”. In: *Communications Engineering* 1.1 (Nov. 25, 2022), pp. 1–15. ISSN: 2731-3395. DOI: `10.1038/s44172-022-00037-0`. URL: `https://www.nature.com/articles/s44172-022-00037-0` (visited on 04/29/2024).

[90] Weifeng Jiang et al. “Dispersion Relation Prediction and Structure Inverse Design of Elastic Metamaterials via Deep Learning”. In: *Materials Today Physics* 22 (Jan. 1, 2022), p. 100616. ISSN: 2542-5293. DOI: `10.1016/j.mtphys.2022.100616`. URL: `https://www.sciencedirect.com/science/article/pii/S2542529322000141` (visited on 04/29/2024).

[91] Zihan Wang et al. “Design of Phononic Bandgap Metamaterials Based on Gaussian Mixture Beta Variational Autoencoder and Iterative Model Updating”. In: *Journal of Mechanical Design* 144.041705 (Feb. 22, 2022). ISSN: 1050-0472. DOI: `10.1115/1.4053814`. URL: `https://doi.org/10.1115/1.4053814` (visited on 04/29/2024).

[92] Zhaxylyk A. Kudyshev et al. “Machine Learning–Assisted Global Optimization of Photonic Devices”. In: *Nanophotonics* 10.1 (Jan. 1, 2021), pp. 371–383. ISSN: 2192-8614. DOI: `10.1515/nanoph-2020-0376`. URL: `https://www.degruyter.com/document/doi/10.1515/nanoph-2020-0376/html` (visited on 04/29/2024).

[93] Yi Zhu and Evgueni T Filipov. “Harnessing Interpretable Machine Learning for Origami Feature Design and Pattern Selection”. In: *arXiv preprint arXiv:2204.07235* (2022).

[94] Zhi Chen et al. “How to See Hidden Patterns in Metamaterials with Interpretable Machine Learning”. In: *Extreme Mechanics Letters* 57 (Nov. 1, 2022), p. 101895. ISSN: 2352-4316. DOI: `10.1016/j.eml.2022.101895`. URL: `https://doi.org/10.1016/j.eml.2022.101895` (visited on 04/26/2024).

[95] Soo Young Lee et al. “Deep Learning-Based Prediction and Interpretability of Physical Phenomena for Metaporous Materials”. In: *Materials Today Physics* 30 (Jan. 1, 2023), p. 100946. ISSN: 2542-5293. DOI: `10.1016/j.mtphys.2022.100946`. URL: `https://www.sciencedirect.com/science/article/pii/S2542529322003443` (visited on 04/29/2024).

[96] Scott Lundberg and Su-In Lee. *A Unified Approach to Interpreting Model Predictions*. Nov. 24, 2017. DOI: `10.48550/arXiv.1705.07874`. arXiv: `1705.07874 [cs, stat]`. URL: `http://arxiv.org/abs/1705.07874` (visited on 04/29/2024). preprint.

[97] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. *Learning Important Features Through Propagating Activation Differences*. Oct. 12, 2019. DOI: `10.48550/arXiv.1704.02685`. arXiv: `1704.02685 [cs]`. URL: `http://arxiv.org/abs/1704.02685` (visited on 04/29/2024). preprint.

[98] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *"Why Should I Trust You?": Explaining the Predictions of Any Classifier*. Aug. 9, 2016. DOI: `10.48550/arXiv.1602.04938`. arXiv: `1602.04938 [cs, stat]`. URL: `http://arxiv.org/abs/1602.04938` (visited on 04/29/2024). preprint.

[99] K. Bertoldi et al. “Flexible mechanical metamaterials”. In: *Nature Reviews Materials* 2 (2017), p. 17066.

[100] Ole Sigmund and Jakob Søndergaard Jensen. “Systematic design of phononic band–gap materials and structures by topology optimization”. In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 361.1806 (2003), pp. 1001–1019.

[101] O. Sigmund. “In Systematic Design of Metamaterials by Topology Optimization”. In: *on Modelling Nanomaterials and Nanosystems, Dordrecht, 2009; Pyrz, R*. Ed. by Iutam Symposium. Rauhe, J. C., Eds. Springer Netherlands: Dordrecht, 2009, pp. 151–159.

[102] O. R. Bilal and M. I. Hussein. “Ultrawide phononic band gap for combined in-plane and out-of-plane waves”. In: *Physical Review E* 84 (2011), p. 065701.

[103] C. Schumacher et al. “Microstructures to control elasticity in 3D printing”. In: *ACM Transactions on Graphics (TOG)* 34 (2015), p. 136.

[104] Christian C Nadell et al. “Deep learning for accelerated all-dielectric metasurface design”. In: *Optics express* 27.20 (2019), pp. 27523–27535.

[105] Ramin Bostanabad et al. “Globally approximate gaussian processes for big data with application to data-driven metamaterials design”. In: *Journal of Mechanical Design* 141.11 (2019).

[106] Tianshuo Qiu et al. “Deep learning: a rapid and efficient route to automatic metasurface design”. In: *Advanced Science* 6.12 (2019), p. 1900128.

[107] David Finol et al. “Deep convolutional neural networks for eigenvalue problems in mechanics”. In: *International Journal for Numerical Methods in Engineering* 118.5 (2019), pp. 258–275.

[108] Peter R Wiecha and Otto L Muskens. “Deep learning meets nanophotonics: a generalized accurate predictor for near fields and far fields of arbitrary 3D nanostructures”. In: *Nano letters* 20.1 (2019), pp. 329–338.

[109] Zhaocheng Liu et al. “Generative model for the inverse design of metasurfaces”. In: *Nano Letters* 18.10 (2018), pp. 6570–6576.

[110] Wei Ma, Feng Cheng, and Yongmin Liu. “Deep-learning-enabled on-demand design of chiral metamaterials”. In: *ACS nano* 12.6 (2018), pp. 6326–6334.

[111] Yang Deng et al. “Neural-adjoint method for the inverse design of all-dielectric metasurfaces”. In: *Optics Express* 29.5 (2021), pp. 7526–7534.

[112] Wei Ma et al. “Probabilistic representation and inverse design of metamaterials based on a deep generative model with semi-supervised learning strategy”. In: *Advanced Materials* 31.35 (2019), p. 1901111.

[113] Liwei Wang et al. “Deep generative modeling for mechanistic-based learning and design of metamaterial systems”. In: *Computer Methods in Applied Mechanics and Engineering* 372 (2020), p. 113377.

[114] Haoran Ren et al. “Three-dimensional vectorial holography based on machine learning inverse design”. In: *Science advances* 6.16 (2020), eaaz4261.

[115] Zhaocheng Liu et al. “Compounding meta-atoms into metamolecules with hybrid artificial intelligence techniques”. In: *Advanced Materials* 32.6 (2020), p. 1904790.

[116] Jiaqi Jiang, Mingkun Chen, and Jonathan A Fan. “Deep neural networks for the evaluation and design of photonic devices”. In: *Nature Reviews Materials* (2020), pp. 1–22.

[117] Wei Ma et al. “Deep learning for the design of photonic structures”. In: *Nature Photonics* 15.2 (2021), pp. 77–90.

[118] Yihao Xu et al. “Interfacing photonics with artificial intelligence: an innovative design strategy for photonic structures and devices based on artificial neural networks”. In: *Photonics Research* 9.4 (2021), B135–B152.

[119] Mahmoud Elzouka et al. “Interpretable inverse design of particle spectral emissivity using machine learning”. In: *arXiv preprint arXiv:2002.04223* (2020).

[120] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems* 27 (2014).

[121] Antonia Creswell et al. “Generative adversarial networks: An overview”. In: *IEEE Signal Processing Magazine* 35.1 (2018), pp. 53–65.

[122] Wei Ma and Yongmin Liu. “A data-efficient self-supervised deep learning model for design and characterization of nanophotonic structures”. In: *Science China: Physics, Mechanics & Astronomy* 63.8 (2020), pp. 1–8.

[123] Jimmy Lin et al. “Generalized Optimal Sparse Decision Trees”. In: *Proc. International Conference on Machine Learning*. 2020.

[124] Guolin Ke et al. “Lightgbm: A highly efficient gradient boosting decision tree”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 3146–3154.

[125] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778.

[126] Zhi Chen et al. “How to see hidden patterns in metamaterials with interpretable machine learning”. en. In: *Extreme Mechanics Letters* 57 (Nov. 2022), p. 101895. ISSN: 2352-4316. DOI: `10.1016/j.eml.2022.101895`. URL: `https://www.sciencedirect.com/science/article/pii/S2352431622001717` (visited on 03/20/2023).

[127] W. J. Zhou and M. N. Ichchou. “Wave propagation in mechanical waveguide with curved members using wave finite element solution”. en. In: *Computer Methods in Applied Mechanics and Engineering* 199.33 (July 2010), pp. 2099–2109. ISSN: 0045-7825. DOI: `10.1016/j.cma.2010.03.006`. URL: `https://www.sciencedirect.com/science/article/pii/S0045782510000824` (visited on 04/10/2023).

[128] Tian-Xue Ma et al. "Energy harvesting of Rayleigh surface waves by a phononic crystal Luneburg lens". en. In: *International Journal of Mechanical Sciences* 227 (Aug. 2022), p. 107435. ISSN: 0020-7403. DOI: `10.1016/j.ijmecsci.2022.107435`. URL: `https://www.sciencedirect.com/science/article/pii/S0020740322003393` (visited on 04/10/2023).

[129] Run Yu et al. "Latticed underwater acoustic Luneburg lens". en. In: *Applied Physics Express* 13.8 (July 2020). Publisher: IOP Publishing, p. 084003. ISSN: 1882-0786. DOI: `10.35848/1882-0786/aba7a7`. URL: `https://dx.doi.org/10.35848/1882-0786/aba7a7` (visited on 04/10/2023).

[130] Martin Lott et al. "Negative index metamaterial through multi-wave interactions: numerical proof of the concept of low-frequency Lamb-wave multiplexing". en. In: *Scientific Reports* 11.1 (Jan. 2021). Number: 1 Publisher: Nature Publishing Group, p. 561. ISSN: 2045-2322. DOI: `10.1038/s41598-020-79572-9`. URL: `https://www.nature.com/articles/s41598-020-79572-9` (visited on 04/10/2023).

[131] Yan Lu and Ankit Srivastava. "Variational methods for phononic calculations". en. In: *Wave Motion* 60 (Jan. 2016), pp. 46–61. ISSN: 0165-2125. DOI: `10.1016/j.wavemoti.2015.08.004`. URL: `https://www.sciencedirect.com/science/article/pii/S0165212515001183` (visited on 07/21/2021).

[132] Ole Sigmund and Jakob Søndergaard Jensen. "Systematic design of phononic band–gap materials and structures by topology optimization". en. In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 361.1806 (May 2003). Ed. by Roger T. Bonnecaze and Gregory J. Rodin, pp. 1001–1019. ISSN: 1364-503X, 1471-2962. DOI: `10.1098/rsta.2003.1177`. URL: `https://royalsocietypublishing.org/doi/10.1098/rsta.2003.1177` (visited on 05/03/2020).

[133] George A. Gazonas et al. "Genetic algorithm optimization of phononic bandgap structures". In: *International Journal of Solids and Structures* 43.18 (Sept. 2006), pp. 5851–5866. ISSN: 0020-7683. DOI: `10.1016/j.ijsolstr.2005.12.002`. URL: `https://www.sciencedirect.com/science/article/pii/S0020768305006281` (visited on 10/29/2023).

[134] Osama R. Bilal and Mahmoud I. Hussein. "Ultrawide phononic band gap for combined in-plane and out-of-plane waves". In: *Physical Review E* 84.6 (Dec. 2011). Publisher: American Physical Society, p. 065701. DOI: `10.1103/PhysRevE.84.065701`. URL: `https://link.aps.org/doi/10.1103/PhysRevE.84.065701` (visited on 10/29/2023).

[135] Yan Lu et al. "3-D phononic crystals with ultra-wide band gaps". en. In: *Scientific Reports* 7.1 (Feb. 2017). Number: 1 Publisher: Nature Publishing Group, p. 43407. ISSN: 2045-2322. DOI: `10.1038/srep43407`. URL: `https://www.nature.com/articles/srep43407` (visited on 11/16/2020).

[136] Florian Maurin et al. "Probability that a band-gap extremum is located on the irreducible Brillouin-zone contour for the 17 different plane crystallographic lattices". en. In: *International Journal of Solids and Structures* 135 (Mar. 2018), pp. 26–36. ISSN: 00207683. DOI:

`10.1016/j.ijsolstr.2017.11.006`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0020768317305103` (visited on 04/24/2020).

[137] Dimitri Krattiger and Mahmoud I. Hussein. "Bloch mode synthesis: Ultrafast methodology for elastic band-structure calculations". en. In: *Physical Review E* 90.6 (Dec. 2014), p. 063306. ISSN: 1539-3755, 1550-2376. DOI: `10.1103/PhysRevE.90.063306`. URL: `https://doi.org/10.1103/PhysRevE.90.063306` (visited on 10/28/2023).

[138] Dimitri Krattiger and Mahmoud I. Hussein. "Generalized Bloch mode synthesis for accelerated calculation of elastic band structures". en. In: *Journal of Computational Physics* 357 (Mar. 2018), pp. 183–205. ISSN: 0021-9991. DOI: `10.1016/j.jcp.2017.12.016`. URL: `https://www.sciencedirect.com/science/article/pii/S0021999117309002` (visited on 08/27/2021).

[139] Eric B. Chin et al. "Spectral extended finite element method for band structure calculations in phononic crystals". In: *Journal of Computational Physics* 427 (Feb. 2021). arXiv: 2009.06913, p. 110066. ISSN: 00219991. DOI: `10.1016/j.jcp.2020.110066`. URL: `http://arxiv.org/abs/2009.06913` (visited on 08/27/2021).

[140] Ankit Srivastava and Sia Nemat-Nasser. "Mixed-variational formulation for phononic band-structure calculation of arbitrary unit cells". In: *Mechanics of Materials* 74 (July 2014). arXiv: 1411.2996, pp. 67–75. ISSN: 01676636. DOI: `10.1016/j.mechmat.2014.03.002`. URL: `http://arxiv.org/abs/1411.2996` (visited on 08/27/2021).

[141] Mahmoud I. Hussein. "Reduced Bloch mode expansion for periodic media band structure calculations". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 465.2109 (July 2009). Publisher: Royal Society, pp. 2825–2848. DOI: `10.1098/rspa.2008.0471`. URL: `https://royalsocietypublishing.org/doi/10.1098/rspa.2008.0471` (visited on 10/28/2023).

[142] Antonio Palermo and Alessandro Marzani. "A reduced Bloch operator finite element method for fast calculation of elastic complex band structures". In: *International Journal of Solids and Structures* 191-192 (May 2020), pp. 601–613. ISSN: 0020-7683. DOI: `10.1016/j.ijsolstr.2019.12.011`. URL: `https://www.sciencedirect.com/science/article/pii/S0020768319304950` (visited on 10/28/2023).

[143] Brian R. Mace et al. "Finite element prediction of wave motion in structural waveguides". In: *The Journal of the Acoustical Society of America* 117.5 (Apr. 2005), pp. 2835–2843. ISSN: 0001-4966. DOI: `10.1121/1.1887126`. URL: `https://doi.org/10.1121/1.1887126` (visited on 10/28/2023).

[144] Brian R. Mace and Elisabetta Manconi. "Modelling wave propagation in two-dimensional structures using finite element analysis". In: *Journal of Sound and Vibration* 318.4 (Dec. 2008), pp. 884–902. ISSN: 0022-460X. DOI: `10.1016/j.jsv.2008.04.039`. URL: `https://www.sciencedirect.com/science/article/pii/S0022460X0800391X` (visited on 10/28/2023).

[145] J. Knap et al. "Adaptive sampling in hierarchical simulation". en. In: *International Journal for Numerical Methods in Engineering* 76.4 (2008), pp. 572–600. ISSN: 1097-0207. DOI:

10.1002/nme.2339. URL: https://doi.org/10.1002/nme.2339 (visited on 03/20/2023).

[146] Kenneth W. Leiter et al. "Accelerated scale-bridging through adaptive surrogate model evaluation". en. In: *Journal of Computational Science* 27 (July 2018), pp. 91–106. ISSN: 1877-7503. DOI: 10.1016/j.jocs.2018.04.010. URL: https://www.sciencedirect.com/science/article/pii/S1877750317313807 (visited on 03/19/2023).

[147] Burigede Liu et al. "A learning-based multiscale method and its application to inelastic impact problems". en. In: *Journal of the Mechanics and Physics of Solids* 158 (Jan. 2022), p. 104668. ISSN: 0022-5096. DOI: 10.1016/j.jmps.2021.104668. URL: https://www.sciencedirect.com/science/article/pii/S0022509621002982 (visited on 03/20/2023).

[148] Ying Shi Teh, Swarnava Ghosh, and Kaushik Bhattacharya. "Machine-learned prediction of the electronic fields in a crystal". en. In: *Mechanics of Materials* 163 (Dec. 2021), p. 104070. ISSN: 0167-6636. DOI: 10.1016/j.mechmat.2021.104070. URL: https://www.sciencedirect.com/science/article/pii/S0167663621002921 (visited on 03/20/2023).

[149] M. Raissi, P. Perdikaris, and G.E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". en. In: *Journal of Computational Physics* 378 (Feb. 2019), pp. 686–707. ISSN: 00219991. DOI: 10.1016/j.jcp.2018.10.045. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999118307125 (visited on 04/24/2020).

[150] Sangryun Lee, Zhizhou Zhang, and Grace X. Gu. "Deep Learning Accelerated Design of Mechanically Efficient Architected Materials". In: *ACS Applied Materials & Interfaces* 15.18 (May 2023). Publisher: American Chemical Society, pp. 22543–22552. ISSN: 1944-8244. DOI: 10.1021/acsami.3c02746. URL: https://doi.org/10.1021/acsami.3c02746 (visited on 08/10/2023).

[151] Romit Maulik et al. "A turbulent eddy-viscosity surrogate modeling framework for Reynolds -averaged Navier-Stokes simulations". en. In: *Computers & Fluids* 227 (Sept. 2021). ISSN: 0045-7930. DOI: 10.1016/j.compfluid.2020.104777. URL: https://doi.org/10.1016/j.compfluid.2020.104777 (visited on 03/20/2023).

[152] Xiaolin Li et al. "A Transfer Learning Approach for Microstructure Reconstruction and Structure-property Predictions". en. In: *Scientific Reports* 8.1 (Sept. 2018). Number: 1 Publisher: Nature Publishing Group, p. 13461. ISSN: 2045-2322. DOI: 10.1038/s41598-018-31571-7. URL: https://www.nature.com/articles/s41598-018-31571-7 (visited on 08/10/2023).

[153] Kaushik Bhattacharya et al. "Model Reduction and Neural Networks for Parametric PDEs". In: *arXiv:2005.03180 [cs, math, stat]* (May 2020). arXiv: 2005.03180. URL: http://arxiv.org/abs/2005.03180 (visited on 05/28/2020).

[154] Zongyi Li et al. “Neural Operator: Graph Kernel Network for Partial Differential Equations”. In: *arXiv:2003.03485 [cs, math, stat]* (Mar. 2020). arXiv: 2003.03485. URL: `http://arxiv.org/abs/2003.03485` (visited on 06/21/2021).

[155] Zongyi Li et al. “Fourier Neural Operator for Parametric Partial Differential Equations”. In: *arXiv:2010.08895 [cs, math]* (May 2021). arXiv: 2010.08895. URL: `http://arxiv.org/abs/2010.08895` (visited on 06/21/2021).

[156] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. OCLC: ocm61285753. Cambridge, Mass: MIT Press, 2006. ISBN: 978-0-262-18253-9.

[157] Ce Liu et al. “Exploring features in a bayesian framework for material recognition”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE. San Francisco, CA, USA: IEEE, 2010, pp. 239–246.

[158] Sean Bell et al. “Material recognition in the wild with the materials in context database”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3479–3487.

[159] Gabriel Schwartz and Ko Nishino. “Visual Material Traits: Recognizing Per-Pixel Material Context”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*. June 2013.

[160] Lavanya Sharan et al. “Image statistics for surface reflectance perception”. In: *JOSA A* 25.4 (2008), pp. 846–865.

[161] Yun-Xian Ho, Michael S Landy, and Laurence T Maloney. “How direction of illumination affects visually perceived surface roughness”. In: *Journal of vision* 6.5 (2006), pp. 8–8.

[162] Kiran S. Bhat et al. “Estimating Cloth Simulation Parameters from Video”. In: *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA ’03. San Diego, California: Eurographics Association, 2003, pp. 37–51. ISBN: 1581136595.

[163] Bernd Bickel et al. “Design and fabrication of materials with desired deformation behavior”. In: *ACM Transactions on Graphics (TOG)* 29.4 (2010), pp. 1–10.

[164] Huamin Wang, Ravi Ramamoorthi, and James F. O’Brien. “Data-Driven Elastic Models for Cloth: Modeling and Measurement”. In: *ACM Transactions on Graphics* 30.4 (July 2011). Proceedings of ACM SIGGRAPH 2011, Vancouver, BC Canada, 71:1–11. URL: `http://graphics.berkeley.edu/papers/Wang-DDE-2011-08/`.

[165] Eder Miguel et al. “Data-driven estimation of cloth simulation models”. In: *Computer Graphics Forum*. Vol. 31. Wiley Online Library. Hoboken, NJ, USA: Wiley Online Library, 2012, pp. 519–528.

[166] Katherine L Bouman et al. “Estimating the material properties of fabric from video”. In: *Proceedings of the IEEE International Conference on Computer Vision*. Sydney, Australia: IEEE, 2013, pp. 1984–1991.

[167] Bohan Wang, Yili Zhao, and Jernej Barbič. “Botanical materials based on biomechanics”. In: *ACM Transactions on Graphics (TOG)* 36.4 (2017), pp. 1–13.

[168] Bin Wang et al. “Deformation capture and modeling of soft objects.” In: *ACM Trans. Graph.* 34.4 (2015), pp. 94–1.

[169] Meekyoung Kim et al. “Data-driven physics for human soft tissue animation”. In: *ACM Transactions on Graphics (TOG)* 36.4 (2017), pp. 1–12.

[170] Hongyi Xu et al. “Interactive material design using model reduction”. In: *ACM Transactions on Graphics (TOG)* 34.2 (2015), pp. 1–14.

[171] Abe Davis et al. “Visual vibrometry: Estimating material properties from small motion in video”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Boston, MA, USA: IEEE, 2015, pp. 5335–5343.

[172] Abe Davis et al. “Visual Vibrometry: Estimating Material Properties from Small Motions in Video”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.4 (2017), pp. 732–745. DOI: `10.1109/TPAMI.2016.2622271`. URL: `https://doi.org/10.1109/TPAMI.2016.2622271`.

[173] Justin G Chen et al. “Structural modal identification through high speed camera video: Motion magnification”. In: *Topics in Modal Analysis I, Volume 7*. Berlin, Germany: Springer, 2014, pp. 191–197.

[174] Yongchao Yang et al. “Blind, simultaneous identification of full-field vibration modes and large rigid-body motion of output-only structures from digital video measurements”. In: *Engineering Structures* 207 (2020), p. 110183. ISSN: 0141-0296. DOI: `https://doi.org/10.1016/j.engstruct.2020.110183`. URL: `http://www.sciencedirect.com/science/article/pii/S0141029618321370`.

[175] Yunus Emre Harmanci et al. “A Novel Approach for 3D-Structural Identification through Video Recording: Magnified Tracking”. In: *Sensors* 19.5 (2019), p. 1229. ISSN: 1424-8220. DOI: `10.3390/s19051229`. URL: `https://www.mdpi.com/1424-8220/19/5/1229`.

[176] Abe Davis et al. “The Visual Microphone: Passive Recovery of Sound from Video”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH)* 33.4 (2014), 79:1–79:10.

[177] Abe Davis, Justin G Chen, and Frédo Durand. “Image-space modal bases for plausible manipulation of objects in video”. In: *ACM Transactions on Graphics (TOG)* 34.6 (2015), pp. 1–7.

[178] Franz Durst, Adrian Melling, and James H Whitelaw. “Principles and practice of laser-Doppler anemometry”. In: *NASA STI/Recon Technical Report A* 76 (1976), p. 47019.

[179] TC Chu, WF Ranson, and Michael A Sutton. “Applications of digital-image-correlation techniques to experimental mechanics”. In: *Experimental mechanics* 25.3 (1985), pp. 232–244.

[180] Hani H Nassif, Mayrai Gindy, and Joe Davis. “Comparison of laser Doppler vibrometer with contact sensors for monitoring bridge deflection and vibration”. In: *Ndt & E International* 38.3 (2005), pp. 213–218.

[181] Nicolaas Bernardus Roozen et al. “Determining radiated sound power of building structures by means of laser Doppler vibrometry”. In: *Journal of Sound and Vibration* 346 (2015), pp. 81–99.

[182] Timothy Emge and Oral Buyukozturk. “Remote nondestructive testing of composite-steel interface by acoustic laser vibrometry”. In: *Materials evaluation* 70.12 (2012), pp. 1401–1410.

[183] Justin G Chen, Robert W Haupt, and Oral Buyukozturk. “The acoustic-laser vibrometry technique for the noncontact detection of discontinuities in fiber reinforced polymer-retrofitted concrete”. In: *Materials evaluation* 72.10 (2014), pp. 1305–1313.

[184] Emanuela Speranzini and Stefano Agnetti. “The technique of digital image correlation to identify defects in glass structures”. In: *Structural Control and Health Monitoring* 21.6 (2014), pp. 1015–1029.

[185] Shih-Heng Tung, Ming-Hsiang Shih, and Wen-Pei Sung. “Development of digital image correlation method to analyse crack variations of masonry wall”. In: *Sadhana* 33.6 (2008), pp. 767–779.

[186] João Feiteira et al. “Monitoring crack movement in polymer-based self-healing concrete through digital image correlation, acoustic emission analysis and SEM in-situ loading”. In: *Materials & Design* 115 (2017), pp. 238–246.

[187] ZhiMin Wu et al. “An experimental investigation on the FPZ properties in concrete using digital image correlation technique”. In: *Engineering Fracture Mechanics* 78.17 (2011), pp. 2978–2990.

[188] Jeffrey D Helm. “Digital image correlation for specimens with multiple growing cracks”. In: *Experimental mechanics* 48.6 (2008), pp. 753–762.

[189] William N MacPherson et al. “Multipoint laser vibrometer for modal analysis”. In: *Applied optics* 46.16 (2007), pp. 3126–3132.

[190] M Martarelli, GM Revel, and C Santolini. “Automated modal analysis by scanning laser vibrometry: problems and uncertainties associated with the scanning system calibration”. In: *Mechanical systems and signal processing* 15.3 (2001), pp. 581–601.

[191] NS Ha, HM Vang, and NS Goo. “Modal analysis using digital image correlation technique: an application to artificial wing mimicking beetle’s hind wing”. In: *Experimental Mechanics* 55.5 (2015), pp. 989–998.

[192] F Trebuňa and M Hagara. “Experimental modal analysis performed by high-speed digital image correlation system”. In: *Measurement* 50 (2014), pp. 78–85.

[193] Dora Foti et al. “Ambient vibration testing, dynamic identification and model updating of a historic tower”. In: *NDT & e International* 47 (2012), pp. 88–95.

[194] Yuriy M. Poplavko. “Chapter 2 - Mechanical properties of solids”. In: *Electronic Materials*. Ed. by Yuriy M. Poplavko. Amsterdam, The Netherlands: Elsevier, 2019, pp. 71–93. ISBN: 978-0-12-815780-0. DOI: `https://doi.org/10.1016/B978-0-12-815780-0.00002-5`. URL: `http://www.sciencedirect.com/science/article/pii/B9780128157800000025`.

[195] Neal Wadhwa et al. “Phase-based video motion processing”. In: *ACM Transactions on Graphics (TOG)* 32.4 (2013), pp. 1–10.

[196] Eero P Simoncelli and William T Freeman. “The steerable pyramid: A flexible architecture for multi-scale derivative computation”. In: *Proceedings., International Conference on Image Processing*. Vol. 3. IEEE. Washington, D.C., USA: IEEE, 1995, pp. 444–447.

[197] Eero P Simoncelli et al. “Shiftable multiscale transforms”. In: *IEEE transactions on Information Theory* 38.2 (1992), pp. 587–607.

[198] Javier Portilla and Eero P Simoncelli. “A parametric texture model based on joint statistics of complex wavelet coefficients”. In: *International journal of computer vision* 40.1 (2000), pp. 49–70.

[199] David J Fleet and Allan D Jepson. “Computation of component image velocity from local phase information”. In: *International journal of computer vision* 5.1 (1990), pp. 77–104.

[200] Neal Wadhwa et al. “Motion microscopy for visualizing and quantifying small motions”. In: *Proceedings of the National Academy of Sciences* 114.44 (2017), pp. 11639–11644.

[201] Martin Alnæs et al. “The FEniCS Project Version 1.5”. In: *Archive of Numerical Software* 3.100 (2015), pp. 9–23.

[202] COMSOL AB. *COMSOL Multiphysics® v5.5*. URL: `https://comsol.com`.

[203] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing In Science & Engineering* 9.3 (2007), pp. 90–95.

[204] Pauli Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: `10.1038/s41592-019-0686-2`.

[205] Greg Turk and Marc Levoy. “Zippered Polygon Meshes from Range Images”. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH ’94. New York, NY, USA: Association for Computing Machinery, 1994, pp. 311–318. ISBN: 0897916670. DOI: `10.1145/192161.192241`. URL: `https://doi.org/10.1145/192161.192241`.

[206] Yixin Hu et al. “Tetrahedral Meshing in the Wild”. In: *ACM Trans. Graph.* 37.4 (July 2018), 60:1–60:14. ISSN: 0730-0301. DOI: `10.1145/3197517.3201353`. URL: `http://doi.acm.org/10.1145/3197517.3201353`.

[207] T.J. Memory, D.P. Thambiratnam, and G.H. Brameld. “Free vibration analysis of bridges”. In: *Engineering Structures* 17.10 (1995), pp. 705–713. ISSN: 0141-0296. DOI: `https://doi.org/10.1016/0141-0296(95)00037-8`. URL: `https://www.sciencedirect.com/science/article/pii/0141029695000378`.

[208] Ali R Kolaini, Walter Tsuha, and Juan P Fernandez. “Spacecraft vibration testing: Benefits and potential issues”. In: *Advances in aircraft and spacecraft science* 5.2 (2018), p. 165.

[209] Sampaio, Regina Augusta and de Souza, Remo Magalhâes. “Vibration Analysis of a Residential Building”. In: *MATEC Web of Conferences* 24 (2015), p. 09007. DOI: `10.1051/matecconf/20152409007`. URL: `https://doi.org/10.1051/matecconf/20152409007`.

[210] Hoss Belyadi, Ebrahim Fathi, and Fatemeh Belyadi. *Hydraulic Fracturing in Unconventional Reservoirs*. Amsterdam, The Netherlands: Elsevier, 2019.

[211] Sierra Sandler, Lauren Alfino, and Mir Saleem. “The Importance of Preventative Medicine in Conjunction with Modern Day Genetic Studies”. In: *Genes & Diseases* 5.2 (June 1, 2018), pp. 107–111. ISSN: 2352-3042. DOI: `10.1016/j.gendis.2018.04.002`. URL: `https://www.sciencedirect.com/science/article/pii/S2352304218300242` (visited on 04/19/2024).

[212] Allison Gates et al. “Screening for the Prevention and Early Detection of Cervical Cancer: Protocol for Systematic Reviews to Inform Canadian Recommendations”. In: *Systematic Reviews* 10.1 (Jan. 2, 2021), p. 2. ISSN: 2046-4053. DOI: `10.1186/s13643-020-01538-9`. URL: `https://doi.org/10.1186/s13643-020-01538-9` (visited on 04/19/2024).

[213] N. Gucunski and R. D. Woods. “Numerical Simulation of the SASW Test”. In: *Soil Dynamics and Earthquake Engineering* 11.4 (Jan. 1, 1992), pp. 213–227. ISSN: 0267-7261. DOI: `10.1016/0267-7261(92)90036-D`. URL: `https://www.sciencedirect.com/science/article/pii/026772619290036D` (visited on 04/19/2024).

[214] Chih-Ping Lin, Chun-Hung Lin, and Chih-Jung Chien. “Dispersion Analysis of Surface Wave Testing – SASW vs. MASW”. In: *Journal of Applied Geophysics* 143 (Aug. 1, 2017), pp. 223–230. ISSN: 0926-9851. DOI: `10.1016/j.jappgeo.2017.05.008`. URL: `https://www.sciencedirect.com/science/article/pii/S0926985116305067` (visited on 04/19/2024).

[215] Jiabei Yuan. “Field Studies Comparing SASW, Beamforming and MASW Test Methods and Characterization of Geotechnical Materials Based on Vs”. In: (Aug. 2011). URL: `http://hdl.handle.net/2152/ETD-UT-2011-08-3981` (visited on 04/19/2024).

[216] Berthy T. Feng et al. “Visual Vibration Tomography: Estimating Interior Material Properties from Monocular Video”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). June 2022, pp. 16210–16219. DOI: `10.1109/CVPR52688.2022.01575`. URL: `https://doi.org/10.1109/CVPR52688.2022.01575` (visited on 04/19/2024).

[217] Ayub Qassim et al. “Corneal Stiffness Parameters Are Predictive of Structural and Functional Progression in Glaucoma Suspect Eyes”. In: *Ophthalmology* 128.7 (July 1, 2021), pp. 993–1004. ISSN: 0161-6420. DOI: `10.1016/j.ophtha.2020.11.021`. URL: `https://www.sciencedirect.com/science/article/pii/S0161642020311167` (visited on 04/19/2024).

[218] Vinay Challa and Hai-Chao Han. “Spatial Variations in Wall Thickness, Material Stiffness and Initial Shape Affect Wall Stress and Shape of Intracranial Aneurysms”. In: *Neurological Research* 29.6 (Sept. 1, 2007), pp. 569–577. ISSN: 0161-6412. DOI: `10.1179/016164107X164193`. pmid: `17535557`. URL: `https://doi.org/10.1179/016164107X164193` (visited on 04/19/2024).

[219] A Gh Podoleanu. “Optical Coherence Tomography”. In: *The British Journal of Radiology* 78.935 (Nov. 2005), pp. 976–988. ISSN: 0007-1285. DOI: `10.1259/bjr/55735832`. URL: `https://www.birpublications.org/doi/abs/10.1259/bjr/55735832` (visited on 04/19/2024).

[220] Brendan F. Kennedy, Kelsey M. Kennedy, and David D. Sampson. “A Review of Optical Coherence Elastography: Fundamentals, Techniques and Prospects”. In: *IEEE Journal of Selected Topics in Quantum Electronics* 20.2 (Mar. 2014), pp. 272–288. ISSN: 1558-4542. DOI: `10.1109/JSTQE.2013.2291445`. URL: `https://ieeexplore.ieee.org/abstract/document/6670078?casa_token=ZhVaNCKHKwgAAAAA:OEyJ1xw_NQstko2W6_gcP34h-IbSV4aj3p03PmoBXutjMQTTvW0v9bexLgj8oyWwgRtsQeCL` (visited on 04/19/2024).

[221] R. A. Bell. “Economics of MRI Technology”. In: *Journal of magnetic resonance imaging: JMRI* 6.1 (1996), pp. 10–25. ISSN: 1053-1807. DOI: `10.1002/jmri.1880060105`. pmid: `8851398`.

[222] Nathalie J. Bureau and Daniela Ziegler. “Economics of Musculoskeletal Ultrasound”. In: *Current Radiology Reports* 4.8 (June 20, 2016), p. 44. ISSN: 2167-4825. DOI: `10.1007/s40134-016-0169-5`. URL: `https://doi.org/10.1007/s40134-016-0169-5` (visited on 04/19/2024).

[223] Neal Wadhwa et al. “Phase-Based Video Motion Processing”. In: *ACM Transactions on Graphics* 32.4 (July 21, 2013), 80:1–80:10. ISSN: 0730-0301. DOI: `10.1145/2461912.2461966`. URL: `https://dl.acm.org/doi/10.1145/2461912.2461966` (visited on 04/20/2024).

[224] M. Proestaki et al. “Modulus of Fibrous Collagen at the Length Scale of a Cell”. In: *Experimental Mechanics* 59.9 (Nov. 2019), pp. 1323–1334. ISSN: 0014-4851. DOI: `10.1007/s11340-018-00453-4`. pmid: `31680700`.

[225] Gregory M. Fomovsky, Stavros Thomopoulos, and Jeffrey W. Holmes. “Contribution of Extracellular Matrix to the Mechanical Properties of the Heart”. In: *Journal of Molecular and Cellular Cardiology*. Special Issue: Extracellular Matrix 48.3 (Mar. 1, 2010), pp. 490–496. ISSN: 0022-2828. DOI: `10.1016/j.yjmcc.2009.08.003`. URL: `https://www.sciencedirect.com/science/article/pii/S0022282809003253` (visited on 04/23/2024).

[226] Jessica E. Wagenseil and Robert P. Mecham. “Vascular Extracellular Matrix and Arterial Mechanics”. In: *Physiological Reviews* 89.3 (July 2009), pp. 957–989. ISSN: 0031-9333. DOI: `10.1152/physrev.00041.2008`. URL: `https://journals.physiology.org/doi/full/10.1152/physrev.00041.2008` (visited on 04/23/2024).

[227] Iain Muntz et al. "The Role of Cell–Matrix Interactions in Connective Tissue Mechanics". In: *Physical Biology* 19.2 (Jan. 2022), p. 021001. ISSN: 1478-3975. DOI: `10.1088/1478-3975/ac42b8`. URL: `https://dx.doi.org/10.1088/1478-3975/ac42b8` (visited on 04/23/2024).

[228] Yunjie Wang, Shahrokh Zeinali-Davarani, and Yanhang Zhang. "Arterial Mechanics Considering the Structural and Mechanical Contributions of ECM Constituents". In: *Journal of Biomechanics*. Cardiovascular Biomechanics in Health and Disease 49.12 (Aug. 16, 2016), pp. 2358–2365. ISSN: 0021-9290. DOI: `10.1016/j.jbiomech.2016.02.027`. URL: `https://www.sciencedirect.com/science/article/pii/S0021929016301798` (visited on 04/23/2024).

[229] Yanping Cao et al. "Tissue-Mimicking Materials for Elastography Phantoms: A Review". In: *Extreme Mechanics Letters* 17 (Nov. 1, 2017), pp. 62–70. ISSN: 2352-4316. DOI: `10.1016/j.eml.2017.09.009`. URL: `https://www.sciencedirect.com/science/article/pii/S2352431617301487` (visited on 04/23/2024).

[230] T.J. Hall et al. "Phantom Materials for Elastography". In: *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 44.6 (Nov. 1997), pp. 1355–1365. ISSN: 1525-8955. DOI: `10.1109/58.656639`. URL: `https://doi.org/10.1109/58.656639` (visited on 04/23/2024).

[231] Barbrina Dunmire et al. "Characterizing an Agar/Gelatin Phantom for Image Guided Dosing and Feedback Control of High-Intensity Focused Ultrasound". In: *Ultrasound in Medicine & Biology* 39.2 (Feb. 1, 2013), pp. 300–311. ISSN: 0301-5629. DOI: `10.1016/j.ultrasmedbio.2012.09.016`. URL: `https://www.sciencedirect.com/science/article/pii/S0301562912005820` (visited on 04/23/2024).

[232] Alexander C. Ogren et al. "Data-Driven Methods for Architected Materials: Data Acquisition, Property Prediction, and Intelligent Design". In: *preparation* (2024).