Complexity Methods in Physics-Guided Machine Learning

Eduardo Brandão

N°d'ordre NNT : 2023STET062

THÈSE de DOCTORAT
DE L'UNIVERSITÉ JEAN MONNET SAINT-ÉTIENNE

Membre de l'Université de LYON

École Doctorale N° 488
École Doctorale Sciences Ingénierie et de Santé

Informatique

Soutenue publiquement le 20/12/2023, par :
Eduardo Brandão

Complexity Methods in Physics-Guided Machine Learning

Devant le jury composé de :

Tommasi, Marc, Professeur, Université de Lille, Président

Amini, Massih-Reza, Professeur, Univ. Grenoble Alpes, Rapporteur
Gallinari, Patrick, Professeur, Univ. Paris Sorbonne, Rapporteur
Colombier, Jean-Philippe, Univ. Jean Monnet Saint-Étienne, Examinateur
Sebag, Michèle, Directrice de Recherche Univ. Paris Saclay, Examinatrice

François, Jacquenet, Professeur, Univ. Jean Monnet Saint-Étienne, Directeur.de thèse
Emonet, Rémi, Professeur, Univ. Jean Monnet Saint-Étienne, Co-directeur.rice de thèse
Habrard, Amaury, Professeur, Univ. Jean Monnet Saint-Étienne, Invité.e

Acknowledgements

This thesis was long in writing, but even longer in not writing it. For a variety of personal reasons, through tragedies and jubilations great and small, this is a PhD that I have been postponing for 20 years. It has been a long, winding road: complex. It is impossible to thank all those who have contributed to its fruition with their support.

I would first like to thank the members of the jury who have accepted to evaluate my work: Massih-Reza Amini and Patrick Gallinari, as well as Jean-Philippe Colombier, Michèle Sebag, Marc Sebban, and Marc Tommasi who accepted the role of examiners, and Amaury Habrard, whom I had the pleasure to invite to the jury as well.

My deepest appreciation goes to my supervisor, François Jacquenet, and my co-supervisor, Rémi Emonet, for their unwavering guidance and support. Their experience and insights were invaluable; this work would not have been possible without them.

This thesis was a complex endeavor. It started three years ago with a suggestion from Marc Sebban, to whom I owe a debt of gratitude for giving me an unexpected second chance to fulfill a dream. This dream then materialized and evolved through many meetings with my supervisors, but also with Amaury Habrard and Marc Sebban, and through many discussions with Jean-Philippe Colombier. Their unofficial supervision was determinant in the outcomes of this work. To all my supervisors, both official and unofficial, I owe a multitude of thanks for their contributions over these three years, but a few stand out. Jean-Philippe, for the discussions, guidance, and support. The "Physics" in the title of this thesis is, in large measure, thanks to you. To Rémi, I am grateful for the razor-sharp insights and for providing clarity where I fumbled. Thank you for so many times having acted as an interpreter between me and myself. To Amaury, I thank him for his apparently infinite time for insightfully answering my questions, scientific and otherwise. I don't know how you manage to always be busy and always have the time. To François, I would like to thank him for his outstanding support and his precious scientific and professional advice. Thank you for accepting being my supervisor. To Marc, I am grateful for his pragmatism, optimism, and deep insights. Thank you for always being able to find something good. I learned a lot from you all.

I would like to thank all my office mates for the great environment, but in particular Paul Viallard, Volodimir Mitarchuk, Antoine Gourru, and Sri Kalidindi. Our discussions were always fruitful, and each of you had an important contribution to what I did. In particular, I would like to thank Paul for all the important advice, Antoine for sharing those cigarettes that I did not smoke with me, and Volodimir, for the birthday cake and for lending me an ear when I needed it the most.

My gratitude extends to Stefan Duffner, for his time, scientific insights, and many fruitful discussions. I would also like to thank Christine Largeron for her encouragement and optimism: I hope that one day I will have the opportunity to offer the same to others as well. To Achim Kempf, I owe a debt of gratitude for his steadfast support in challenging times. To Edward Vrscay, I would

like to express my thanks for his kindness and encouragement.

Finally, I would like to thank my family: my father, for showing me that a single person can be many things; my mother, for never giving up. My brother Filipe, who is my other half: thank you for our lives. To my wife Lucy, who was there for me when I needed it the most, often at great personal sacrifice: thank you for everything. Thank you for encouraging me to do unreasonable things.

I would like to thank my children: Édouard, Isabelle, and Clara. You are the light of my eyes, and I share all this with you. I wish that someday you will also be able to share something that deeply matters to you with someone you love as dearly as I do you.

Finally, I would like to thank my grandparents, Joaquim e Hermínia, Eduardo e Eufélia, and my godparents, António e Lurdes, who made me who I am today. I dedicate this thesis to the memory of them:

Meus queridos, esta tese dedico-a a vocês. Obrigado por tudo, obrigado pelo vosso amor. Perdoem-me por não a terem visto, mas não fui capaz de a fazer mais cedo. Obrigado por tudo. Tudo farei para que os meus filhos conheçam o que é o amor. Sei que estão sempre comigo, mas fazem-me muita falta.

Contents

Chapter 1

Introduction

1.1 Physics-guided machine learning

This thesis is about physics-guided machine learning [Che+18; Jia+21; RPK19; Um+20; Wil+20], a nascent field that has garnered significant attention in the last few years. The story, like a creation myth, begins with an original sin: the implication that the integration of physics and machine learning is a novel endeavor. In reality, physics — the most fundamental of the natural sciences — has been guiding and inspiring machine learning, as well as other scientific disciplines since its inception. These have borrowed methods and techniques (and also some of the foremost experts), sometime wholesale, in a reciprocal relationship that persists to this day. In return, Machine Learning has contributed with methods and techniques which have been employed to solve complex problems more efficiently, uncover hidden dynamics by data, and even search for new physics.

But from this mild original sin of bearing a not perfectly accurate name, comes redemption. The existence of this new field, this new name, forces the question of how to make the cross-pollination between the two disciplines explicit, and congregates efforts and collaborations.

From the many questions that may occur, we shall start with the following: how to best integrate explicit physical knowledge into a machine learning model in order to facilitate or guide learning?

To answer it, we must first define what we mean by "physical knowledge". Is it a constraint connecting physical quantities? A differential equation describing their evolution? Perhaps a conservation law expressing some fundamental symmetry in Nature?

And once we integrate this knowledge into our machine learning model, another question arises: how much of the model is physics, and how much is machine learning?

Estimating quantity of knowledge using Shannon's entropy To tackle these questions, we turn to the Bayesian interpretation of probability, which considers probability distributions p_i as descriptions of states of knowledge, expressed in terms of the relative likelihood of the outcomes of a given experiment i. The more certain one is about the outcomes of such an experiment, the more knowledge one has.

In 1957, Edward T. Jaynes proposed to estimate "quantity of knowledge" using a measure [Jay57a; Jay57b] that was devised in the context of communication by Claude Shannon in 1948 [Sha48] to

estimate the number of relays necessary for Bob to receive a message from a certain emitter Alice[1].

Shannon's measure became known as Entropy, so the story goes, because the foremost physicist and polymath John von Neumann told Shannon in conversation that *"in the first place, your uncertainty function has been used in statistical mechanics under that name. In the second place, and more importantly, nobody knows what entropy really is, so in a debate you will always have the advantage."* [SW93].

Entropy, the measure that has been winning Shannon all the discussions since 1948, is simply expected surprise (estimated as $\log 1/p_i$):

$$H = -\sum_i p_i \log p_i \tag{1.1.1}$$

The more one knows about the outcomes of an experiment, the less surprising these would be on average. In this sense, then, Shannon entropy can be seen as estimating the amount of information (rather lack of, i.e. *uncertainty*) in a state of knowledge described by a probability distribution p_i.

There are other measures of uncertainty and complexity, as we shall see in Chapter 2. But Shannon's possesses a number of interesting properties, namely that it is the sole measure, up to a multiplicative constant, that satisfies a few reasonable properties that one would assign "information" or "quantity of knowledge" [Sha48; Khi57].

But as Shannon himself mentions in his seminal work, that single-handedly created an entire scientific discipline, Information Theory, axiomatic well-posedness is not the most important aspect of his ideas – possible applications are. And Information theory, which at heart was devised to engineer communication systems that can reliably pass messages between Alice and Bob, has seen applications for beyond its original scope, in a host of domains — amongst which physics and machine learning.

Physical entropy: Clausius, Boltzmann, and Gibbs John von Neumann's name suggestion is not, of course, a mere jest. Physics has long had a measure called *Entropy*, which emerged in the context of Thermodynamics, the branch of physics that studies heat and energy transfer, in the mid-19th century, with Rudolf Clausius. It measures the amount of thermal energy in a system that is unavailable for doing mechanical work. For a system at temperature T undergoing a process from states A to B where an infinitesimal quantity of heat dQ_{rev} is exchanged reversibly at each step with its surroundings, the change in entropy ΔS is given by:

$$\Delta S = \int_{A\to B} \frac{dQ_{\text{rev}}}{T}. \tag{1.1.2}$$

Admittedly, this quantity seems rather far from the measure in eq. (1.1.1). To understand von Neumann's suggestion, we move a bit forward in time to the late 19th century, where Boltzmann and Gibbs gave the concept of entropy a statistical sense. The Boltzmann-Gibbs entropy is defined as the logarithm of the number W of microstates —specific arrangements of the microscopic particles in a system, characterized by their positions and momenta—that are compatible with a given macrostate — which is defined by observable macroscopic properties like temperature and pressure. As can be seen today carved on Boltzmann's tombstone in Vienna, this is

$$S = k \log W, \tag{1.1.3}$$

[1] Shannon was aware of the many possible applications of his ideas. See e.g. Weaver's introduction in [SW63]

where k is Boltzmann's constant[2]. Since the formalization effort of Statistical Physics, entropy has since colloquially been used as a measure of disorder of thermodynamic states, in the sense that the higher it is, the more microstates are accessible to a thermodynamical system — and hence the system is seen as more disordered.

Information theory, thermometers, and calorimeters But this interpretation was generally seen as that – an interpretation – of a measure that is anchored in an experimental reality much like mass, length, or volume. Indeed *"Entropy is a definite physical quantity that can be measured in the laboratory with thermometers and calorimeters."*, as the prominent physicist George Uhlenbeck protested when Jaynes showed that statistical physics can be formalized in purely Information Theoretic terms [Jay78] using Shannon's measure of uncertainty.

Let us make clear how remarkable Jaynes' claim is: thermodynamics, which was conceived as a physical theory anchored in experiment (thermodynamics was built phenomenologically *from* experiment), can actually be seen as a theory of *our* state of knowledge about a certain physical situation. Rather than a strictly physical theory, that describes physical quantities, statistical physics describes how states of knowledge compatible with (described by) macroscopic variables of state evolve, and result in new macroscopic variables.

To emphasize Uhlenbeck's point, recall how to actually measure the entropy of a certain quantity, a mole, say, of a certain substance. We begin by cooling a mole of the substance to (near) zero Kelvin, where it has (near) zero entropy by the third law of thermodynamics, and then slowly heat it up in small incremental steps to a final state A, allowing it to reach equilibrium at every step, while using the calorimeter and the thermometer to measure $\frac{C_{p_k}}{T}$, where C_{p_k} is the molar heat capacity at a constant pressure at each step. Since for a reversible step $\frac{dS}{dT} = \frac{C_{p_k}}{T}$, adding up all these quantities we obtain, in the limit of infinite steps, the entropy of the final state.

One then understands Uhlenbeck's protestation: Jaynes claims that a major branch of physics is actually based on states of knowledge, which are subjective; how then can this statement be compatible with the *fact* that you and I can go in a lab and measure entropy using a thermometer and a calorimeter using the procedure above and agree on the measured quantity?

Gibbs' resolution of Gibbs paradox Jayne's and Uhlenbeck's claims are reconcilable, and they are both correct. A beautiful and clear answer to this question was given by Jaynes [Jay92], repeating an argument by Gibbs, which we now present. Consider n_1 and n_2 moles of two different but ideal gases inside adjacent volumes V_1 and V_2 separated by a membrane, such that $\frac{V_1}{V_2} = \frac{n_1}{n_2}$, and at the same temperature and pressure. If we remove the membrane and the gases are allowed to mix, the entropy of the mixture increases, by standard thermodynamics, as it there is no way to unmix the gases without effecting external changes (making work). But if the membrane separates two equal gases, removing changes nothing. The entropy remains the same. This is what is commonly known as "Gibbs paradox".

The reason for the entropy increase cannot lie in the actual physical microstate of the gases, Jaynes states, repeating an argument by Gibbs. Because it is no more difficult to bring the two different gases to its initial positions state than it is to bring the two equal gases to their original positions. But there is a difference when we consider thermodynamic state, rather than physical state, where instead of having to bring molecules to the original location, we just have to bring the

[2]The constant is usually and in the rest of the thesis denoted k_B, where "B" stands for Boltzmann, but denoting it as such for posterity in one's own tombstone would be unfitting.

same number of molecules of the same gas to the *same side whence they came*. From a thermodynamic point of view, this new state is indistinguishable from the original state. Hence, Jayne argues, the reason for the entropy increase lies in the *knowledge* that the gases are made from different particles. It is this knowledge that defines the state, which can thus change, independently of the actual physical setting, as our knowledge does as well.

This is compatible with Uhlenbeck's method, as two researchers having the same knowledge would agree in their measurement of entropy: the same amount of work would be required by either to return the mix to its original thermodynamical state – as defined by their knowledge. It takes considerably more work to return a mix of two different gases to their original thermodynamic state — hence the higher entropy.

Statistical physics can in this sense be seen simultaneously as an objective theory, and as based on subjective states of knowledge. The subjective quantity measuring uncertainty of those states of knowledge can be objectively measured!

Thermodynamical laws being expressed in terms of entropy, we have that in this extreme case, an entire physical theory can be regarded wholesale as the evolution of knowledge (or uncertainty, i.e., lack thereof) about a certain physical situation, the quantity of which can be measured using a calorimeter and a thermometer.

The previous discussion is hopefully a convincing argument that defining what constitutes physical knowledge to incorporate in a machine learning model, and measuring how much of it there is, is far from straightforward. We presented an information-theoretical measure, entropy, that can be used to estimate the amount of knowledge in a probability distribution, as suggested by Jaynes. And then we saw that an entire branch of physics can actually seen in information-theoretical terms, as drawing conclusions from the evolution of this quantity of knowledge — which, remarkably, can actually be measured in the laboratory using calorimeters and thermometers!

We are not always in this extreme case, of course. We do not always have the luxury to measure the quantity of knowledge in physical information, such as Newton's laws, for example, using lab instruments like thermometers and calorimeters. Nor will we be able to do so for the Swift-Hohenberg equation modeling Rayleigh-Bénard convection, which we will examine in detail in this thesis. Neither is it common that physical knowledge comes neatly formulated in terms of a probability distribution of the states of the system that is being considered. Much more often this knowledge is rather given in terms of other quantities which bring about some implicit knowledge about the physical situation. How can this be quantified and used?

Physical knowledge about laser-matter interaction in the Swift-Hohenberg equation The first topic in this thesis belongs in Surface Engineering, more specifically, laser-matter interaction. The problem that we addressed was that of learning the relationship between laser parameters such as fluence, for example, and the nanometer-scale topography modifications that the laser-matter interaction induces on the surface of Nickel.

As discussed in detail in Chapter 3, the number of topography/laser parameter pairs that we have access to for training in order to learn this relationship is extremely (and fundamentally) low, in the order of the tens of data points (one single scanning electron microscope image per dynamics). How can one then learn anything other than the most simplistic relationship using Machine Learning methods alone?

One cannot. But thankfully, the knowledge that we have about Nickel and laser-matter interaction, as well as the controlled experimental conditions, is extensive. For example, we know the atomic mass of Nickel, we know its melting temperature, etc. We also know that laser-matter

interaction, in spite of its extraordinary complexity, can be approximately described as a sequence of different stages: in a first stage, energy is transferred from the electromagnetic field to the matter; in a second stage, the molten material follows a hydrodynamic process roughly governed by Navier-Stokes type equations, until in a final stage it solidifies again. The Navier-Stokes equations, although much too complex to admit an analytical solution (even numerical solution are challenging), still constitute a massive simplification regarding the actual physical process taking place, which we will simplify even further. Symmetry and physical considerations discussed in detail in Chapter 4 motivate choosing an even simpler equation, called the *Swift Hohenberg equation*, that approximately models the first hydrodynamics stage of the laser-matter interaction, given a number of simplifying assumptions that are satisfied in the experimental setting.

If one would be able to integrate this physical knowledge to guide learning, then one could learn a complex and *useful* relationship between laser parameters and observed topographies, even with few data. Unfortunately, solving these equations to model dynamics requires initial and boundary conditions, to which we do not have access.

We know all of this, but unfortunately (i) it is unclear how to use it, and (ii) it is unclear whether it is enough to successfully learn under such constrained conditions. The reason behind the first difficulty is clear: the knowledge that we would like to integrate is not given *explicitely* about the quantities that we wish to relate, nor is there a known relationship between these quantities and those present in the Swift Hohenberg equation. To make up for this fact, one would typically use data, which we do not have; in our case, we project the dynamics into a lower-dimensional space, which considerably simplifies the problem.

Knowledge about the evolution of knowledge in the Swift-Hohenberg equation To understand the second difficulty, note that what we would like to ideally do with this partial differential equation is to integrate enough knowledge in the learning process to make up for the fact that we are trying to learn a full dynamical process[3] on the basis of a single Scanning Electron Microscope image. This is of course hopelessly complicated, and generally impossible to do.

Crucially, the Swift Hohenberg equation does not just bring knowledge regarding the hydrodynamics of the situation. It also encapsulates the knowledge that only the Fourier modes in the vicinity of a certain critical mode are important in the long term dynamics. It contains not only physical information, but also information about how our knowledge of this information evolves. That this harks back to the statistical physics discussion above is no coincidence, because the Swift Hohenberg equation is really a statistical model in disguise! (cf. Sec.3.5.2).

Why is this crucial? Because in order to learn with a single data point, we must have a model (think a Hilbert space with basis functions $\{\phi_n\}$) in which the pattern field can be expressed in terms of a single basis element. So the physical knowledge that we have is going to be useful to us precisely because it tells us that initial conditions (within reason) are not important to select which spatial frequencies we will end up observing. The uncertainty that we have on the spatial frequencies is bound to decrease. In this case, we shall show, the relationship can be learned even in the case of extremely low data.

The data-equivalent of physical knowledge A certain equation or a certain symmetry bring knowledge about a certain problem, as it can replace a great number of data in certain situations – but possibly not others. The knowledge that we integrate using physical information, as measured in

[3]To be precise, we do not really need the full process, just enough that we can use it to predict pattern features based on laser parameters. See Chapter 3 for details.

"quantity of data equivalent", so to speak, depends on the relationship between physical information and the data.

As we just saw, the Swift-Hohenberg equation allows dramatic simplification, to the point that it can effectively replace an infinity of data *in a learning problem that, as we shall see, consists mostly in learning spatial-frequencies*. But the underlying complexity of the problem is still there: it is just a matter of asking the right questions. If instead of focusing on the spatial frequencies of the patterns we wish to learn the actual value of the field at a certain point, the knowledge about the behavior of the frequencies is not going allow the same, dramatic, simplification.

This motivates the following questions: how can we meaningfully compare the knowledge in data and the knowledge in a some physical information, in an integrated way? Can they be given the same units, so to speak, and compared?

Minimum description length principle as a principle of model selection The second part of this thesis (cf. Chapter 4) focuses on the Minimum Description Length principle [Ris78], a principle of probabilistic model selection proposed by Jorma Rissanen in the late 1970s. This principle can be seen as a formalization of Occam's razor, often stated as "entities should not be multiplied beyond necessity" or more simply, "the simplest explanation is the best one." In the context of the Minimum Description Length principle, one equates descriptions and probability distributions (cf. Sec. 4.3.2) we seek the probabilistic model family that provides the shortest description length — which is seen as a measure of complexity — for the data, comprising the description length of the model itself. The goal is to find the simplest model that adequately explains the data, thereby embodying the essence of Occam's razor.

In the early 2000's [Ris01] Rissanen proposed a way to measure the description length, in which the two complexities are neatly separated into two terms, which are measured in the same "units": essentially, model family complexity is measured by tallying up the likelihood of all possible data (cf. Section 4.3.5). This can be seen as a measure of the explaining power of a model family, using the behavior of elements of the model family with respect to data.

The data-equivalent of knowledge for a task via the Minimum description length principle The minimum description length principle thus provides a unified way to measure the "quantity of knowledge" in data (its description length) and in the model family (the sum of the maximum likelihoods that elements of the model family can assign arbitrary data). Unfortunately, as we shall see in Chapter 4, this conceptually beautiful approach is riddled with technical complications that limit its scope.

Moreover, albeit unifying the "quantity of knowledge" in data and in the model family (which one may think as formalizing physical information in the Bayesian sense above), the minimum description length principle, in this form, does not quantify the "quantity of knowledge" *for a specific problem*. To take a physical example, data that comes from ocean observations should be seen as simple by a good physical model of ocean circulation. Conversely, a good model of ocean circulation would be able to replace a considerable amount of ocean circulation data.

In this second part of our work in Chapter 4, we were able to address some of these difficulties by proposing a new Minimum Description Length measure of complexity which measures the complexity of a model class (our physical knowledge) in terms of its behavior with respect to classification-relevant and classification-irrelevant data, in the context of neural network classification problems. For differentiable Neural Network models trained by gradient descent to minimize

classification loss[4], we show that when learning is good (in the sense that it generalizes well), then the model is good as well (in the sense that it can replace a lot of data).

Complexity in Physics and Machine Learning In the first part of the works that constitute the core of this thesis, we incorporated knowledge of laser-matter interaction in a particular experimental setting, and also, crucially, how the knowledge of this situation will evolve (much like in the statistical physics sense of Jaynes) to learn the relationship between laser parameters and nanopatterns in the extremely low data regime. In the second part, we sought a formalism in which data and knowledge, as formalized via probabilistic model families, are in the same footing. We proposed a modification of this formalism that depends on the task. As we have just argued, understanding this relationship is essentially finding the data equivalent of a certain physical model. This will eventually allow one to gauge how much data and of what type will be needed to tackle a certain problem, about which we have a certain physical knowledge.

1.2 Thesis organization and contributions

1.2.1 Complexity, full of sound and fury

A number of measures of complexity have been proposed in the literature. In this chapter, we introduce, compare and contrast a number of such measures relevant to the context of patterns formed self-organization, which we illustrate with the nanopatterns observed in Nickel that will be the main topic of the next chapter.

1.2.2 Learning Complexity to Guide Light-Induced Self-Organized Nanopatterns

In the context of physics-guided machine learning, the task is often to solve the inverse problem by incorporating physical information to guide learning. In many applications, however, this is not feasible due to severe constraints in the quantity of data, lack of access to initial conditions, incompleteness of the physical model, among others. In this case, a combination of methods is in order, and even so success is not guaranteed. In the case of self-organization processes, we show that because complexity decreases in time, we can tackle the inverse problem under severe data constraints. This, in turn, provides deep physical insights with respect to the underlying physical process, and opens the door to automatic exploration of the solution space.

Contributions

We solve the inverse problem for femtosecond laser-induced nanopatterns on monocrystalline (100) Nickel by integrating physical information in the form of the Swift-Hohenberg equation, which is an archetypal model of type I-s pattern formation and a model of Rayleigh-Bénard convection. We do so under severe data constraints, and in the *fundamental* absence of dynamical data, which allows us to validate the hydrodynamic nature of the novel patterns observed in Nickel upon irradiation, and opens the door to automatically exploration of pattern space. In unpublished work, we also propose a novel physical mechanism for pattern formation, based on the two temperature model, extending the work in [Rud+20] to the first picoseconds before thermalization.

[4]More generally, Lipchitz continuous, although the paper focused on Neural Networks.

This line of research combines methods and techniques in physics and in machine learning. As research at the intersection between two fields of research, it could be perceived by the community as "jack of all trades, master of none". Instead, the paper led to two publications, each focusing on the Physics and on the Machine Learning aspect:

- Brandão, Eduardo, Anthony Nakhoul, Stefan Duffner, R. Emonet, Florence Garrelie, Amaury Habrard, François Jacquenet, Florent Pigeon, Marc Sebban, and Jean-Philippe Colombier. "Learning Complexity to Guide Light-Induced Self-Organized Nanopatterns." Physical Review Letters 130, no. 22 (2023): 226201.
- Brandão, Eduardo, Jean-Philippe Colombier, Stefan Duffner, Rémi Emonet, Florence Garrelie, Amaury Habrard, François Jacquenet, Anthony Nakhoul, and Marc Sebban. "Learning PDE to model self-organization of matter." Entropy 24, no. 8 (2022): 1096.

1.2.3 Is my neural Net Guided by the MDL principle?

The answer is "yes". The Minimum description length principle, which was introduced by Rissanen in the late 1970s, can be seen as a formulation of Occam's razor: from a set of competing hypothesis, choose the simplest. Specifically in Machine Learning: choose the model family that provides the simplest description of the data, comprising the model itself. In spite of the intuitive attractiveness of this formulation, there remains the conceptual difficulty in how to measure the complexity of the model.

Contributions

We propose a formulation of the MDL principle that addresses the aforementioned conceptual difficulty, by formulating it in terms of signal and noise in the training data as defined by the task itself. By formulating MDL in this way, we are also able to resolve a number of technical difficulties regarding its application, and predict the distribution of the local Jacobian spectrum of Neural Network classifiers trained according to the MDL principle. Following closely the experimental setting in [Zha+17], we show experimentally that Jacobian spectra for different model types and different data sets do agree with predictions, and thus provide indication that Neural Networks do follow the MDL principle.

This line of research led to a publication studying the dynamics of learning of Neural Networks, in the following conference paper:

- Brandão, Eduardo, Stefan Duffner, Rémi Emonet, Amaury Habrard, François Jacquenet, and Marc Sebban. "Is My Neural Net Driven by the MDL Principle?." In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 173-189. Cham: Springer Nature Switzerland, 2023.

Chapter 2

Entropies and measures of complexity

2.1 Introduction: complexity of measures of complexity

How complex is a square? Is it more or less complex than a circle? The latter seems to be intuitively simpler but both are balls, respectively for the l_1 and for the l_2 distances on the plane. How complex is a cloud on a summer day? Is it more or less complex than a fractal that looks like a cloud on a summer day but which is generated using a simple deterministic program on a computer? The latter seems to be intuitively simpler but the former is arguably also generated by another simple deterministic program: physics. How complex is the set of all sequences that can be written by a computer? How complex is the set of all sequences that can be generated by physics? How do they compare?

To summarize, the previous paragraph is intended to show that complexity is a deep cross-disciplinary topic. I should know that because I wrote it. But it seems to be intuitively much more complex than the summary that was just provided. For once, it is much longer, running at 688 characters including spaces, whereas the summary is only 93 characters long. But although Einstein's *On the electrodynamics of moving bodies* [Ein05] is 31 pages long in its original version, it seems unreasonable to claim that it it less complex than 32 pages of the letter "o". Why?

We shall provide an answer to some of these questions in the sequel, but the fact remains that complexity is complex to define. Indeed, taking into account the previous discussion, it should be no surprise that to date, no general and widely accepted means of measuring complexity exists. On the contrary, the number of measures of complexity are legion ([Edm97] alone cites 386 different measures of complexity), and their justification often lies on their ability to produce intuitive results.

There is some regularity to these measures, however, as most can be seen as approximations or estimates of a few fundamental types. [Ben18] for example, divides measures of complexity in those which are based on function and those based on structure. In [Gra12], the difficulty in providing a single, all-encompassing definition of complexity is seen as fundamental. There cannot be a single measure of complexity, not because we have not been able to find it, but because it depends on what the observer is interested in computing. This subjectiveness is regarded as fundamental, much as in Quantum Mechanics, for example. A taxonomy based on task is proposed instead.

In this work, for the purpose of systematization, we shall adopt Lloyd's taxonomy of complexity [Llo01] which also adopts a task-based approach, but further organizes measures of complexity into three main axes: (i) difficulty of description, (ii) difficulty of generation, and (ii) degree of organization.

We shall be interested in measures of complexity which are applicable to self-organization and pattern formation, more specifically to the recently discovered ultrafast laser-induced nanopatterns described in [Rud+20; Nak+21a] and which, we showed recently by using physics-guided machine learning techniques in [Bra+23; Bra+22], can be modeled by the Swift-Hohenberg equation, which is a maximally symmetric model of pattern formation.

The goal is not to provide a definitive measure of complexity in this case, which we already argued is impossible to provide. Rather, our investigation will follow Bennet, who functionally defines complexity in this setting as "whatever increases when something self-organizes" [Sha01]. In our case, "something" are the laser-induced nanopatterns, and which we shall use as test-bed for appropriate measures of complexity.

The program is as follows: we begin by introducing Lloyd's taxonomy and provide a general overview of the complexity measures used in this thesis, integrating them within this system. At the basis of all the measures used in this thesis, are the notions of Shannon entropy and Kolmogorov complexity. We shall strive to provide, as much as possible, a self-contained primer that introduces the notions that will be used in the fundamental notions of complexity: Shannon entropy and Kolmogorov complexity, which we hope will be useful to physicists. These will serve as a basis to introduce and examine a number of reasonable measures of complexity found in the literature, propose a few generalization that we believe to be relevant for our context, and apply them to the 2-dimensional patterns described in [Abo+20] and to the Swift-Hohenberg equation.

One of the main difficulties that we encounter is that the measures above turn out to be measures of "disorder" or "information" rather than measures of "complexity" (in the intuitive sense). This is an important problem that we shall examine in Section 2.4. Regarding this point, we shall originally propose a compression-based measure of complexity, which we call *Intensive Lempel-Ziv complexity*. To our knowledge, this is the first measure of this kind that is not distributional based.

Another important problem in the literature is the absence of a canonical measure of complexity for the evolution of multidimensional field. To address this difficulty, we propose a novel measure of disorder and associated complexity *Taylor entropy* and *Taylor complexity*, the former being a natural generalization of *Permutation entropy* to the multidimensional case and which takes local information into account.

2.2 Three axes of complexity

Lloyd measures complexity along three different axes [Llo01]: (i) *difficulty of description*, (ii) *difficulty of generation*, and (iii) *degree of organization*.

2.2.1 Axis 1: Complexity as difficulty of description

The first axis of Lloyd's taxonomy measures complexity by difficulty of description. It is the most fundamental of the three axes, as it lays the foundations for the other two.

Shannon entropy A description is essentially a way to identify an object uniquely amongst other objects. If we attempt to do so based on the relative likelihood of observing the object in

comparison to those in a previously agreed upon set, then the only reasonable notion (cf 2.3.3) is Shannon entropy. In this approach, which can be regarded as either probabilistic or combinatorial in nature [Kol65], the measure that we obtain can be seen as quantifying the degree of uncertainty in predicting an outcome or, conversely, as a measure of how much of that uncertainty we lose when performing a measurement. That quantity is the information, and Shannon entropy if a measure thereof.

Assigning a precise meaning to uncertainty can be done in different ways. In Shannon's case, information theory is essentially an engineering theory[1], and uncertainty is measured in number of relays. As described by Shannon, entropy is the answer to the following question: how many relays does a receiver need on average in order to identify a received message amongst a set of pre-defined messages, each with a possibly different probability of being observed? Other ways to assign a meaning to this uncertainty lead to generalizations of Shannon entropy: Rényi entropy and Tsallis entropy, which we present, respectively, in Sec. 2.3.5 and Sec. 2.3.4.

However uncertainty may be quantified: it is uncertainty *with respect to what*? We can apply Shannon's ideas to different types of messages, or even to different aspects of the same message[2], which will in turn correspond to uncertainty with respect to the relative likelihoods of different things. For example, as we shall illustrate by applying them to different SEM images of the laser-induced nanopatterns on the surface of monocrystalline Nickel (which will be the topic of Chapter 3), the power and phase spectrum of an images. But the same ideas can also be applied to the letter frequencies of the English language, etc. The significance of this measure of complexity is clearly different in each of these cases but an engineering theory of relays should not concern itself with the meaning of the messages. Each message should simply be identified by said relays, irrespective of their individual nature. In the words of Shannon [Sha48]:

> *The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. Frequently the messages have meaning; that is they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem. The significant aspect is that the actual message is one selected from a set of possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design.*

Kolmogorov complexity But one can of course think of many situations where semantic aspects of communication or correlations "*according to some system with certain physical or conceptual entities*" are important. One such case is *individual* message complexity. Intuitively, some messages are clearly more complex than others. First, we note that quantifying the complexity of *individual* messages cannot be meaningfully done in the context of Shannon's engineering theory of relays: for the number of relays that takes to identify a message does not depend on the content of the message itself.

Defining complexity of an individual message cannot be done in the void. Doing so requires that we appeal to the relationship between the message and "*certain physical or conceptual entities*". These need to be sufficiently general that they can be used in as many situations as possible:

[1] An engineering theory with far-reaching conceptual implications, which Shannon certainly did not miss!
[2] E.g. some function of the message.

ideally, we would like to be able to compare the complexities of the human genome and Beethoven's Symphony No. 9.

The idea is to describe the object based on some sufficiently general ("Universal") way. Doing so, we are arguably identifying the object "intrinsically". If we measure difficulty of identification by the length of this description, then we obtain what is known as *Kolmogorov complexity* or *algorithmic complexity*, an idea that was developed independently by Solomonov [Sol64], Chaitin [Cha69] and Kolmogorov [Kol65]: the shortest description length amongst all computable descriptions (cf. Sec. 2.3.1).

Similarly to Shannon entropy, we can measure Kolmogorov complexity of different aspects of the same object. Deciding which aspect is important for the problem at hand is of course crucial.

A major difficulty with this approach is the incomputability of Kolmogorov complexity. The most common way around this problem is to restrict the universality of the law in some fashion. If we do so dramatically and in a specific case (but still recovering Kolmogorov complexity asymptotically), leads to Lempel-Ziv (cf. 2.6.5). Choosing to do so in a more flexible way, we obtain Stochastic complexity (cf. 4.3.5). We shall examine both approaches, the latter with special care as it led to one recent publication where we propose a generalization measure for neural network classifiers that is based on the Minimum Description length principle (where description length is known as Stochastic complexity).

As we shall see, complexity in the statistical sense and in the individual message sense are related: as we shall see, Shannon entropy can be recovered as the expectation of Kolmogorov complexity in a number of interesting cases. Remarkably, there is no such result for either Rényi entropy nor Tsallis entropy [Tei+11].

Order vs. complexity The problem with both Shannon's and Kolmogorov's approaches is that they are arguably measures of order or information rather than measures of complexity. Shannon entropy being a measure of uncertainty implies that it is maximal for probability assignments which, based on the *principle of indifference* assign the same probability to every outcome. Most of us would arguably describe a set of n samples from the corresponding distribution as being "disordered" rather than complex.

Similarly, if an individual message cannot be described succinctly using a sufficiently general "universal" way, then most of us would call it disordered rather than random. To see this, consider that *most objects must be complex*, as there is simply not enough space at the bottom: assume that an n-bit object can be compressed by at least k bits. There are 2^n objects with n bits and the number of compressed objects is $1 + 2 + 4 + \cdots + 2^{n-k} = 2^{n-k+1} - 1$. The probability that an object can be compressed by more than k bits is thus approximately 2^{-k+1}. This probability quickly becomes exceedingly small. With $k = rn$, where r is the compression rate, one sees that the probability that a 100 bit string can be compressed by more than ten percent ($r = 0.9$) is only 2^{-8}: 99.61% of all 100 bit strings cannot be compressed by more than 10%.

Hence, an object that cannot be simplified is in the majority of the possible strings, and we would characterize the great majority of strings as being disordered rather than complex. Hence, as Shannon entropy, complexity in the Kolmogorov (incompressibility) sense seems to correspond to disorder rather that intuitive complexity.

Strings that one would characterize as complex show some sort of regularity, but are not perfectly ordered. They seem to be neither totally ordered nor totally disordered, the maximum of complexity occurring somewhere in the middle. For a given measure of disorder D, this is most simply constructed by taking $D(1 - D)$, but we shall examine in Sec. 2.4 a more sophisticated

measure, where $(1 - D)$ is replaced with a measure of dissimilarity[LMC95].

2.2.2 Axis 2: Complexity as difficulty of generation

The second axis of Lloyd's taxonomy of complexity is difficulty of generation. Although we shall not be treating this approach in this thesis, we mention what is arguably its a key example, Bennet's Logical Depth [Ben88]. Simply put, logical depth also looks at objects in terms of universal descriptions. But rather than measuring complexity by how long the minimal description is, one looks at how much *time* that description needs to be computed. Like others before it, this approach is physically motivated: extant physical objects must be produced efficiently, otherwise in all likelihood they would not be observed. In terms of self-reproducing objects the argument is even more pertinent, as competition by resources means that objects that we observe must be more efficient than others at producing the same structure. If there would be two possible approaches, the one taking more time would quickly overwhelm the one that took longer. As Kolmogorov complexity, logical depth is incomputable.

2.2.3 Axis 3: Complexity as degree of organization

The third axis of Lloyd's taxonomy of complexity is the degree of organization. On the one hand, this can be measured by the relationship between the different parts of the object, which is typically measured in terms of either mutual information or self-correlation (cf. Sec 2.6.4) or its Kolmogorov analogue (cf. Normalized compression distance for a computable approximation (2.6.12)). On the other hand, one can examine the degree of organization in the Dynamical system sense. In our illustration, we focus on a measure of chaos, the chief example thereof being Metric entropy or Kolmogorov-Sinai entropy of a dynamical system. Here too, incomputability, motivates the introduction of approximations of this measure, such as *Permutation entropy* (cf. Sec. 2.5.3), which has the added advantage that can be applied even in presence of noise. As we shall see, it does not, however, generalize straightforwardly to a multidimensional setting. To address this difficulty, we shall define and apply novel measure of organization applicable in the multidimensional case, that, since it depends on the spacial gradients, is particularly relevant in the case where the field results from a dynamical process.

2.3 Measures of complexity: axis 1

We shall now define and illustrate, when possible, using the SEM images discussed in Chapter 3 a number of measures of complexity.

2.3.1 Kolmogorov Complexity

The structure of the following argument can be summarized as follows: (i) Turing machines provide a model for computability: each Turing machine can be associated with a certain (partial) function, which is called *computable*. The Church-Turing thesis asserts that functions that can be effectively calculable by humans are computable. (ii) there are Turing machines that can simulate the behavior of any other Turing machine. These machines are special as they provide a universal mean of computation. (iii) since I can humanly generate any finite string x, so can a universal Turing machine T. The length of the smallest self-delimited prefix input to the machine that generates x is

called the Kolmogorov complexity (iv) This length of this input is at most a (potentially enormous) constant from the Kolmogorov complexity with respect to another machine, the constant being the cost of simulating the other machine (which is a partial function, and hence can be simulated by T; asymptotically for very long strings, the two lengths coincide.

This is a brief overview of Kolmogorov complexity, that we present for completeness. For an extensive exposition, please refer to [LV13].

Turing Machine as a model of computability

A *Turing Machine* [Min19] T consists of a finite program called the *finite control*, which can move and write symbols on a linear list of cells called a *tape* using a *head*. The finite control consists of a finite number of states $\tilde{Q} = \{q_0\} \cup Q$, with $Q = \{q_1, \ldots, q_n\}$, and the head can write either 0, 1, or B on the tape. Computation proceeds in discrete time steps starting from an initial configuration in which the machine is in the start state q_0, and the head is positioned over a special "start cell." To the right of such a cell, there is a certain number of cells containing a sequence of 0 and 1, called the *input*. All other cells on the tape are blank (B). Computation consists of the following actions:

1. The machine scans the tape for the symbol directly under it.
2. The machine overwrites it with either $\{0, 1, B\}$.
3. The machine shifts the head one cell left or right.
4. The machine assumes a state q_i from Q.

Items 2 and 3 are called *operations*.

The machine is constructed in such a way that it behaves according to a set of rules, consisting of quadruples $(q_t, s_t, o_{t+1}, q_{t+1})$, where t designates the current time step, $q_t \in \tilde{Q}$ is the current machine state, s_t is the symbol under scan, $o_{t+1} \in \{0, 1, B, L, R\}$ is the operation to perform, and q_{t+1} is the new state after the operation. The machine T is deterministic, in the sense that $(o_{t+1}, q_{t+1}) = f(q_t, s_t)$. Importantly, the machine can *halt* if it reaches a state (o, q) for which there is no associated rule. Deterministic Turing Machines can be identified by this mapping.

Computability Let $\phi : A \to B$ be a function. Then ϕ is a *partial function* if, for each $x \in A$, either $\phi(x) \in B$, in which case we say that "x is a *value* of ϕ"; or $\phi(x)$ is *undefined*. If every $x \in A$ is a value of ϕ, then ϕ is called a *total* function. Otherwise, it is called a *strictly partial* function [Min19]. Every Turing Machine can be associated with a partial function. We can associate each Turing Machine with a partial function from n-tuples of integers onto $\mathbb{N}$ in the following sense: we take the input as $x^n = (x_1, \ldots, x_n)$, where $x_i \in \{0, 1\}^*$ and write it on the tape in self-delimiting form $1^{l(x_1)}0x_1 1^{l(x_2)}0x_2 \cdots 1^{l(x_n)}0x_n$; and take the output as the maximal binary string (bordered by blank cells) written on the tape when the machine halts. Partial functions that can be associated with Turing machines in this way are called *partial computable*. If T halts for every input, we call the partial function that is associated with it *total* or *computable*.

Kolmogorov complexity as description length by a Universal Turing Machine We call a Universal Turing Machine (UTM) a Turing machine that can simulate the behavior of any other Turing machine, in the following sense: it takes as input a description of another Turing machine and an input for that machine and simulates its execution. The length of the smallest self-delimited prefix input to the machine that generates x is called the Kolmogorov complexity

Definition 1. *Kolmogorov complexity, denoted $K(x)$, is a measure of the complexity or information content of a string x relative to a Turing machine. It is defined as follows in terms of a Universal Turing Machine:*

$$K(x) = \min\{|p| : T(p) = x\}$$

where $K(x)$ is the Kolmogorov complexity of x relative to the Universal Turing Machine T, p is the shortest program that generates x when run on T, and $|p|$ is the length of program p.

Relative to a Universal Turing Machine T, Kolmogorov complexity differs from one Turing machine to another by at most a constant. This constant is essentially the length of the program that encodes the other Turing machine (which is a partial function, and hence can be simulated by T). Asymptotically, as the length of the strings increases, the cost of simulating the other machine, which is constant, becomes less important. For very long strings, asymptotically, the two lengths coincide. One can define Kolmogorov complexity analogs of information theoretic quantities, using string concatenation in the place of joint distributions.

Unfortunately, Kolmogorov complexity is not computable, which motivates a number of approximations that we discuss further below:

Theorem 2.1. *The complexity function K is not computable; moreover, any computable lower bound for K is bounded from above.*

Proof. Assume that k is a computable lower bound for K which is not bounded from above. Then for any m, we can effectively find a string x such that $K(x) > m$, since its lower bound is computable and not bounded by above. Consider now f defined as

$$f(m) = \text{the first discovered string } x \text{ such that } k(x) > m$$

Note that by definition, $K(f(m)) > m$; on the other hand, f is a computable function and therefore $K(f(m)) \leq K(m) + O(1)$, since the output of f on a universal Turing machine can be simulated by simulating f at a fixed cost, together with the cost of simulating the input; and the latter is lower-bounded by the Kolmogorov complexity of the input. But since $K(m) \leq |m| + O(1)$, we have that $m \leq |m| + O(1)$ which is impossible since a natural number is not smaller than the length of its binary representation ($n \leq \lceil \log_2(n) \rceil$) □

2.3.2 Boltzmann-Gibbs entropy

In the context of statistical mechanics, one typically considers *ensembles*, that is copies of the same system, that are compatible with some measurement or property, called a *macrostate*. The precise definition of each copy of the system specifying all the positions and momenta $\vec{x}_i, \vec{p}_i$ of each of its composing particles[3] $i = 1 \dots N$ is called a *microstate*.

To each macrostate one can associate a probability distribution, which assigns to each of the microstates j a certain probability p_j[4]. In the *canonical ensemble*, the system is in thermal equilibrium with a heat bath at temperature T and the probability of the system being in microstate j with energy E_j is given by

$$p_j = \frac{1}{Z} e^{-E_i/k_B T},$$

[3]The system is usually supposed composed of N monoatomic particles.

[4]More exactly, a *Liouville function* $W_N(\vec{x}_1, \vec{p}_1, \dots, \vec{x}_N, \vec{p}_N, t)$, which gives the probability density in the phase space of the system

where k_B is the Boltzmann constant and Z is a normalization constant called the partition function. Then we define the *Gibbs entropy* as

$$S_G = -k_B \sum_i p_i \ln p_i. \tag{2.3.1}$$

If the total energy of the system is specified, i.e. the macrostate is specified by the temperature and energy, all p_j are equal to $1/W$, the number of possible microstates with energy E (the ensemble of which is called the *microcanonical ensemble*. In this case, Boltzmann's entropy S is defined as a measure of the number of microstates W accessible to the system in the given macrostate:

$$S_B = k_B \ln W \tag{2.3.2}$$

. Specifically, Jaynes showed in [Jay65] that in the canonical ensemble, the difference in Gibbs entropy over a reversible path coincides with that obtained via Clausius' definition 1.1.2 (and hence in the case of an ideal gas with uniform density and temperature and no inter-particle forces, with the difference in Boltzmann entropy).

This allows him to show the second law: the argument is as follows. The canonical distribution minimizes Gibbs' entropy for a given mean energy, over all distributions compatible with it. Recalling that at the canonical distribution, Gibbs entropy and Clausius' coincide, it follows that over all ensembles with the same mean energy, Gibbs entropy S_G is a lower bound of Clausius' S_C

$$S_G \leq S_C.$$

Now if a system begins at complete thermal equilibrium and hence macroscopic quantities can be represented by the canonical distribution, experimental and Boltzmann entropy coincide $S_G(A) = S_C(A)$. If one moves the system adiabatically from A to B using e.g. a piston (no heat flows), then by Liouville's theorem, because the energy is constant, the phase space distribution does not change, and hence neither does Gibbs' entropy $S_G(B) = S_G(A)$.

Now wait until the system is allowed to come once more to equilibrium with a new experimental entropy $S_C(C)$. At this new state, the system is no longer necessarily represented by the canonical distribution. But since the mean energy did not change, the new ensemble is still compatible with the initial mean energy, which was lower bounded by the Gibbs entropy $S_G(A) \leq S_C(C)$. But since $S_G(A) = S_C(A)$ by assumption, we have for adiabatic paths

$$S_C(A) \leq S_C(C),$$

which is the Second Law of thermodynamics.

To relate this derivation with the discussion in the introduction regarding entropy and knowledge, note that in going from A to C, the knowledge of the mean energy did not change (by construction and Liouville's theorem): hence, the Gibbs entropy remains the same. But how well that mean energy actually characterizes the system did: we lost a great deal of knowledge regarding the microstates as the system evolved, and this is reflected in the non-decrease in Clausius' entropy[5].

[5] See also [Jay92] for an interesting thought experiment with isotopes that shows that this entropy is objective but still dependent on a state of knowledge.

2.3.3 Shannon entropy

Definition 2 (Shannon entropy)**.** *Let X be a discrete random variable with alphabet $\mathcal{X}$ and probability mass function $p(x) := \Pr(X = x), x \in \mathcal{X}$. Then we define the Shannon entropy (or simply entropy) as [CT12]*

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log p(x) = \mathbb{E}_X\left[\log\frac{1}{p(x)}\right] \tag{2.3.3}$$

where the logarithm is taken in base 2 and we set $0\log 0 = 0$ throughout this work.

Interpretations of Shannon entropy Since the logarithm is a strictly increasing function, $\log\frac{1}{p(x)}$ can be seen as a measure of the surprise of outcome x, and entropy can be interpreted as the expectation of this measure of surprise. Another fruitful interpretation of the Shannon entropy can be given in terms of the expected length of encoding (intuitively, a description) of samples from X. In fact, the Kraft-Macmillan inequality [CT12] show that the expected length for any uniquely decodable code C of a random variable X over an alphabet of size D is greater than or equal to $H_D(X)$, the Shannon entropy calculated in base D, with equality holding iff $D^{-l_i} = p_i$.

Axiomatic definition of Shannon entropy

Shannon Entropy can be characterized axiomatically. There are several ways to do so, but we shall present two, the first due to Shannon himself [Sha48] who actually states (Section 6, page 50):

> *This theorem, and the assumptions required for its proof, are in no way necessary for the present theory. It is given chiefly to lend a certain plausibility to some of our later definitions. The real justification of these definitions, however, will reside in their implications.*

and the second due to Khinchin [AK57]. There are similar characterization of Tsallis entropy and Rényi entropy [Abe00]. Other axiomatic characterizations of Shannon entropy exist: in [BFL11], for example, it is shown that Shannon entropy is the only measure of information loss that composes well, mixes well, and is robust.

Khinchin axioms Khinchin proves that three properties uniquely define entropy up to a multiplicative constant in a theorem that we adapt below:

Theorem 2.2. *Given a discrete random variable X with distribution $p_1, p_2, \cdots, p_n$. Let $H(p_1, \cdots, p_n)$ be continuous with respect to all its arguments and have the following properties:*

1. *For any given n, the function $H(p_1, \cdots, p_n)$ takes its largest value for X uniformly distributed $p_i = \frac{1}{n}, \forall i$*
2. *$H(X, Y) = H(X) + H(Y|X)$*
3. *$H(p_1, \cdots, p_n, 0) = H(p_1, \cdots, p_n)$*

Then

$$H(p_1, p_2, \cdots, p_n) = -\lambda \sum_{i=1}^{n} p_i \log p_i, \tag{2.3.4}$$

where λ is a positive constant.

Note that we define H as a continuous function of a distribution measuring the amount of "choice", "disorder" or "ignorance". The first property states then that among all distributions, the one with equally probable states is maximally "disordered". The third property states that in estimating ignorance, we do not consider impossible events.

Recalling that $p(x, y) = p(x)p(y|x)$[6] and that $H(Y|X) := \sum_{i=1}^{m} p(x_m)H(p(y_1|x_m), \cdots, p(y_n|x_m))$, or, more concisely, $H(Y|X) := \sum_{i=1}^{m} p(x_m)H(Y|X = x_m)$, we can write the second property as

$$\begin{aligned} H\left(p(x_1)p(y_1|x_1), \cdots, p(x_1)p(y_n|x_1), \cdots, p(x_m)p(y_1|x_m), \cdots, p(x_m)p(y_n|x_m)\right) \\ = H(p(x_1), \cdots, p(x_n)) + \sum_{i=1}^{m} p(x_i)H(p(y_1|x_i), \cdots, p(y_n|x_i)) \end{aligned}$$

Proof. The property is essentially that of the uniqueness of the logarithm. With $H(n)$ the entropy of uniform random variable with n elements, and $H(n, 0)$ the entropy of random variable with $n + 1$ elements, n of each equally probable and the remaining impossible, from the first and third property, we have that $H(n) = H(n, 0) \leq H(n + 1)$, so $H(n)$ is a non-decreasing function of n. □

Shannon's axioms Shannon proves uniqueness of entropy based on similar properties. Rather than deriving monoticity, Shannon requires it axiomatically. The spirit of the properties, however, is similar:

1. H should be continuous in the p_i.
2. If all the $p_i = \frac{1}{n}$ are equal, then $H(n)$ should be a monotically increasing function of n.
3. If a choice is to be broken into two successive choices, the original H should be the weighted sum of the individual values of H

To see how the last property, which is known as the *coarse-graining property* because it prescribes how to the entropy at two different scales is calculated, is the same as the second property above, consider a pair of random variables (first choice, second choice), which have the following joint distribution for picking left/right.

XY	r	l	X
r	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{2}$
l	0	$\frac{1}{2}$	$\frac{1}{2}$

[6] In this section, $p(x)$ is short for $p_X(x)$ or $P(X = x)$ and it denotes the probability that the random variable X takes the value x. Abusing notation, $p(x, y)$ is short for $p_{XY}(x, y)$, etc.

Shannon implicitly assumes hypothesis 3 above, and writes

$$H(\frac{1}{2},\frac{1}{3},\frac{1}{6},0) = H(\frac{1}{2},\frac{1}{2}) + \frac{1}{2}H(\frac{2}{3},\frac{1}{3})$$

In the language of property 2 above, we have $H(X) = H(1/2, 1/2)$ and $H(Y|X) = \frac{1}{2}H(1) + \frac{1}{2}H(\frac{2}{3},\frac{1}{3})$. So Shannon really is assuming that $H(1) = 0$, which squares up quite well with the interpretation on H as a measure of *choice*. The proof is now quite similar and is based on the uniqueness of the logarithm. The interested reader may find a proof in [Die69].

2.3.4 Tsallis entropy

Tsallis entropy was introduced in [Tsa88] as a generalization of the Boltzmann-Gibbs entropy:

Definition 3 (Tsallis entropy)**.** *Given a discrete random variable X with probability mass function p_i and q any real number, the* Tsallis entropy *of X is defined as*

$$S_q(X) = \frac{k}{q-1}\left(1 - \sum_i p_i^q\right) \tag{2.3.5}$$

where q is a real parameter called entropic index and k is a positive constant.

Importantly, in the limit $q \to 1$, $k = k_B$ the Boltzmann constant we recover the Gibbs entropy, as can be shown straightforwardly using l'Hôpital's rule:

$$\begin{aligned}\lim_{q\to 1} k_B \frac{1-\sum_i p_i^q}{q-1} &= \lim_{q\to 1} k_B \frac{-\sum_i p_i^q \log p_i}{1} \\ &= -k_B \sum_i p_i \log p_i\end{aligned}$$

Tsallis entropy is a measure of disorder. To see this, note that[7] when when $q = 2$, we have $S_2(X) = 1 - \sum_i p_i^2$. This is the negative of the probability that independently identically distributed (i.i.d.) random variables X and Y are not in the same state, that is $S_2(X) = P(X \neq Y)$. To provide an intuition as to in what sense this a measure of disorder, we provide two examples, one in the dynamical system sense, the other in a sense that is more applicable to an SEM image.

Disorder in the sense of dynamical system Let X be a random variable describing a scalar field at time t resulting from the evolution from some slightly perturbed initial conditions u by some physical process. Then if Y is an i.i.d random variable, the probability $P(X \neq Y)$ can be approximated by letting a large number N of pairs of fields evolve from slightly perturbed initial conditions for a long time, and counting the number of times the pairs do not coincide. If the evolution is very sensitive to initial conditions, then the number of coincidences n is small, and $S_2(X) \approx \frac{N-n}{N}$ is large. For larger integer $q > 0$ the intuition is the same, but we now only count coincidences if they occur in q samples at the same time, a stricter condition. Specifically, if a state has low probability, then the number of times we observe q coincidences is going to be exceedingly small. Large q thus weights proportionally more coincidences in high probability states.

[7] For simplicity, we work in units such that $k_B := 1$

Disorder in the sense of collection of samples Consider an 8-bit SEM image. Each pixel can be in one of 256 states $u_0, \ldots, u_{255}$. Then 2.3.5 can be approximated by sampling a large number N of times q -tuples from the image and counting the number of times they are in the same state. If an SEM image is uniform, then we expect this number to be large. If the image appears to be disordered, on the other hand, we would expect this number to be small.

Disorder is disorder of something The first definition of disorder depends on symmetries, for example: we count coincidences after projecting the field onto a certain feature space. The second notion of disorder is tied to the definition of state. An image that has maximal disorder when considering states to be pixel values can still be perceptually ordered: coincidence of states in this case corresponds to coincidence of *shapes* such as stripes or dots.

Axiomatic definition of Tsallis entropy Similarly to Shannon entropy, Tsallis entropy can be derived axiomatically [Abe00]. The main difference with respect to the Shannon entropy is that Tsallis entropy is no longer necessarily extensive. Specifically, given independent random variables X, Y we have

$$S_q(X,Y) = S_q(X) + S_q(Y) + (1-q)S_q(X)S_q(Y) \tag{2.3.6}$$

The system is called *superadditive* or *subadditive* if respectively, $q < 1$ or $q > 1$.

Determining the entropic index q As we saw above q determines the sensitivity for low probability events for a given problem. It is typically determined by fitting data to theoretical distributions [GL04]. In the absence of a theoretical distribution, there is no standard procedure for setting it. For image thresholding applications it is commonly set by eye [Ram+16]. In the same paper, the authors set q by maximizing q-redundancy $R_q(X) = 1 - \frac{S_q(X)}{S_q^{max}}$, where $S_q^{max} = k\frac{1-\Omega^{1-q}}{q-1}$ is the maximum Tsallis entropy for a random variable with Ω states. In the same lines, q has also been set in the context of anomaly detection in such a way that it maximizes the difference between normal and abnormal behavior [ATT12].

Importantly, there is a relationship between entropic index q and sensitivity to initial conditions at the onset of chaos [GT04].

2.3.5 Rényi entropy

Entropy does not need to have the coarse-graining property: this is precisely the motivation in [Sha06], where in page 49, in the context of Khinchin's axiomatic approach, Rényi entropy is introduced via a relaxation of the coarse-graining property (axiom 2 in Khinchin's characterization).

Definition 4 (Rényi entropy of order α)**.** *The* Rényi entropy *of order $\alpha \geq 0, \alpha \neq 1$ of a discrete random variable X is defined as [Rén+61]*

$$S_\alpha(X) = \frac{1}{1-\alpha} \log \left(\sum_{i=1}^{n} p_i^\alpha \right) \tag{2.3.7}$$

where p_i is the probabilities of the ith outcome of X and the logarithm is commonly taken to be base two.

When $\alpha \to 0$, Rényi entropy becomes the log-cardinality, also known as Hartley entropy of X. When $\alpha \to 1$ we obtain the Shannon Entropy, as can be shown using l'Hôpital's rule:

$$\begin{aligned}\lim_{\alpha\to 1} \frac{\log \sum_i p_i^\alpha}{1-\alpha} &= \lim_{\alpha\to 1} -\frac{\sum_i p_i^\alpha \log p_i}{\sum_i p_i^\alpha} \\ &= -\sum_i p_i \log p_i,\end{aligned}$$

and when $\alpha = 2$ we obtain collision entropy (often just called "Rényi entropy"), which measures the probability that a pair of samples from X are identical. As in the case of the Tsallis entropy, Rényi entropy can thus be interpreted in terms of probabilities of coincidences, specifically α-coincidences (on a log scale).

Note that unlike Tsallis entropy, Rényi entropy is additive. If X, Y are independent random variables

$$S_\alpha(X,Y) = S_\alpha(X) + S_\alpha(Y) \tag{2.3.8}$$

2.4 Complexity versus disorder

It has been argued in [LMC95] that the entropic measures above capture the notion of disorder rather than complexity.

Complexity should be a product of entropy and a measure of *disequilibrium*, which is essentially some form of distance to the maximum entropy distribution such as the Jensen-Shannon Divergence $D_{JS}(\cdot,\cdot)$:

$$D_{JS}(X,Y) = \frac{1}{2}\left(D_{KL}(X||M) + D_{KL}(Y||M)\right) \tag{2.4.1}$$

where $M = \frac{1}{2}(X+Y)$ and $D_{KL}(\cdot||\cdot)$ is the Kullback-Leibler divergence.

In [Ros+07] the authors extend this notion to define a statistical complexity measure that is:

1. Able to grasp essential details of the dynamics.
2. Intensive, i.e., does not depend on the size of the system.
3. Capable of discerning among different degrees of periodicity and chaos.

This is achieved by defining $C_{JS}(X)$, the *intensive statistical complexity* of random variable X:

$$C_{JS}(X) = \alpha \frac{D_{JS}(X, X^*)H(X)}{H(X^*)} \tag{2.4.2}$$

where X^* is such that p_{X^*} is the maximum entropy distribution (the uniform distribution in this case) and α is a constant that sets complexity to $[0,1]$. Since the properties of KL-divergence imply $D_{JS}(X,Y) = H(M) - \frac{H(X)+H(Y)}{2}$, replacing in (2.4.2), we obtain

$$C_{JS}(X) = \alpha\left(\frac{H(\frac{X+X^*}{2})H(X)}{H(X^*)} - \frac{H(X)}{2} - \frac{H^2(X)}{H(X^*)}\right)$$

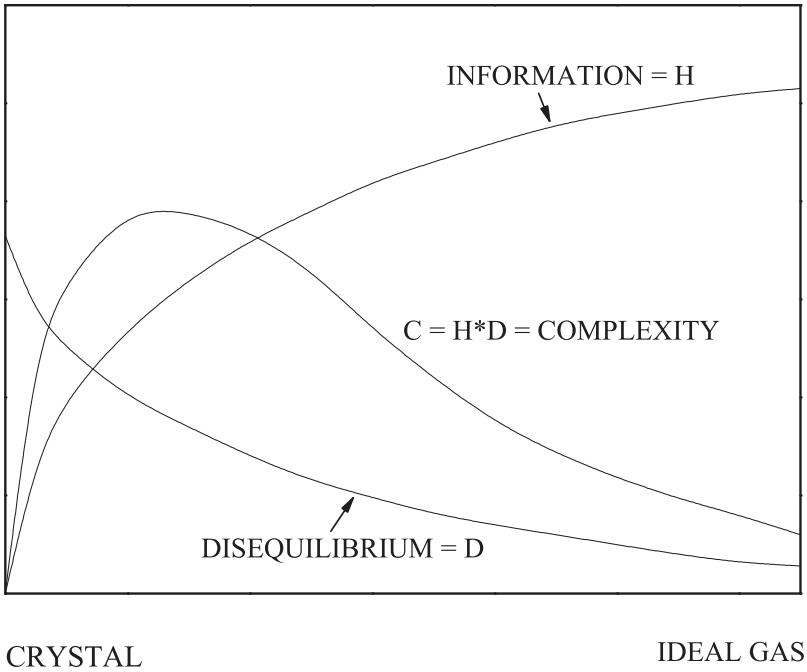


Figure 2.1: The definition of complexity presented in [LMC95] with archetypal examples "crystal" (simple but ordered) and "gas" (simple but disordered) on the x-axis.

Note the $H(\frac{X+X^*}{2})$ term that complicates dependency in what would otherwise be essentially, up to axes translations, H^2. The intensive character of the measure of complexity comes via division by $H(X^*)$ and the fact that the distribution being considered (the distribution of ordinal patterns used to compute permutation entropy is the case of [Ros+07])) is an intensive quantity.

We shall use this notion to define intensive statistical complexities associated with Rényi and Tsallis entropy, which we call, respectively, Rényi and Tsallis complexity. To do so, we replace the Jensen-Shannon divergence with their respective Rényi and Tsallis analogues, replacing the KL divergence in D_{JS}, respectively, by the Rényi divergence [VH14]

$$D_\alpha(P \parallel Q) = \frac{1}{\alpha - 1} \log \left(\sum_x P(x)^\alpha Q(x)^{1-\alpha} \right) \tag{2.4.3}$$

and q-Divergence [Tsa88]

$$D_q(P \parallel Q) = \frac{1}{q - 1} \left(\sum_x P(x)^q Q(x)^{1-q} - 1 \right). \tag{2.4.4}$$

We shall also, in Section 2.6.5, originally propose *Intensive Lempel-Ziv complexity* (cf. eq. (2.6.13)), based on a compression-based measure of disequilibrium.

2.5 Measures of complexity: axis 3

2.5.1 Complexity of a dynamical system

Before we begin by defining a complexity measure we provide a short dynamical systems primer for completeness. A concise but very clear treatment can be found in [San96], who cites [GH13] for details.

Dynamical systems primer

A dynamical system is a set of functions given by an equation expressing how they change. This change can be either discrete or continuous:

Definition 5 (Continuous-time dynamical system)**.** *An n-dimensional continuous-time autonomous smooth dynamical system is defined by the differential equation*

$$\dot{x} = F(x) \tag{2.5.1}$$

where $\dot{x} = \frac{dx}{dt}$, $x(t) \in \mathbb{R}^n$ is the state vector *at time t, and $F : U \to \mathbb{R}^n$ is a C^r function ($r \geq 1$) on an open set $U \subset \mathbb{R}^n$.*

Definition 6 (Flow)**.** *We say that the vector field F generates the flow $f : U \times \mathbb{R} \to \mathbb{R}^n$, where $f^t(x) := f(x, t)$ is C^r defined*

$$\dot{f^t(x)} = F(f^t(x)), \quad \forall x \in U, t \in \mathbb{R} \tag{2.5.2}$$

Definition 7 (Discrete-time dynamical system)**.** *A C^r map $f : U \to \mathbb{R}^n$ on an open set $U \subset \mathbb{R}^n$ defines an n-dimensional discrete-time autonomous dynamical system by the state equation*

$$x_{t+1} = f(x_t) \tag{2.5.3}$$

where x_t is the state of the system at time t and f maps to the next state x_{t+1}.

Complexity of a dynamical system

In the context of a dynamical system, one may wish to quantify the system's complexity, particularly its sensitivity to initial conditions. To illustrate this concept, consider a communication problem: Alice wants to send Bob a string deterministically generated by a dynamical function f from a certain initial condition. All Alice needs to do is send the initial conditions, because assuming both Alice and Bob know f, Bob can then generate an arbitrarily long string from it that will exactly match hers. However, depending on the sensitivity of f to initial conditions, and given that Alice and Bob cannot communicate the initial condition with infinite precision in finite time, their generated orbits from slightly differing initial conditions may diverge. One can then ask: how frequently must Alice resend the initial conditions to Bob to ensure that their generated orbits never diverge by more than a given tolerance?

If this frequency is calculated in bits per symbol, and under a number of restrictive assumptions (chiefly ergodicity), the answer to this question defines what is known as the Kolmogorov-Sinai (KS) entropy (rate) h_μ of the dynamical system $\dot{y} = f$:

Definition 8 (Kolmogorov-Sinai Entropy)**.** *Let $(X, \mathcal{F}, \mu)$ be a probability space and $T : X \to X$ be a measure-preserving transformation. Given a finite measurable partition α of X, the Kolmogorov-Sinai entropy $h_\mu(T)$ is defined as:*

$$h_\mu(T) = \sup_\alpha \lim_{n\to\infty} \frac{1}{n} H\left(\bigvee_{k=0}^{n-1} T^{-k}\alpha\right)$$

where H is the Shannon entropy of the partition $H(\beta) = -\sum_{b\in\beta} \mu(b)\log\mu(b)$, and $\bigvee_{k=0}^{n-1} T^{-k}\alpha$ denotes the refinement of the iterated pullback of α. [GP83]:

This measure of the sensitivity of the system to initial conditions is unfortunately incomputable in a practical setting. One then uses the *Lyapunov exponents*, which are (approximately) computable without ergodicity requirements:

Definition 9 (Lyapunov Exponents)**.** *Given a dynamical system $y(k+1) = F(y(k))$, the Jacobian matrix J_k at step k is defined as $J_k = \frac{dF}{dy}\big|_{y=y(k)}$. The i-th Lyapunov exponent λ_i is defined by:*

$$\lambda_i = \lim_{L\to\infty} \frac{1}{L} \log \sigma_i$$

where σ_i is the i-th singular value of the matrix product $J_L \cdots J_1$, with J_l evaluated along the orbit $y(k), k = 1, \ldots, L$. [ER85]

Moreover the following result relates the two measures for ergodic systems:

Proposition 1 (Pesin's Entropy Formula)**.** *For an ergodic system, the Kolmogorov-Sinai entropy h_μ can be expressed as:*

$$h_\mu = \sum_{\lambda_i > 0} \lambda_i$$

where the sum is taken over all Lyapunov exponents λ_i that are greater than zero. [Pes77]

The Lyapunov exponents can be approximated using the *local* versions: the ith Lyapunov exponent of the system at L time steps is the logarithm of the ith singular value of the product of Jacobians of all the intermediate steps from x. The local Lyapunov exponents can be computed from local data as described in [ABK92], and their sum yields a measure of local Kolmogorov-Sinai entropy (via a local Pesin's type formula [ABK91]).

The authors in [CV13] estimate λ_1 using a technique introduced in [Ego+00], which essentially determines the $L = 1$ Lyapunov exponents, involving only a singular value decomposition of the Jacobian matrix, and is straightforward to calculate. Interestingly, the authors superimpose the magnitude of λ_1 on a field at a given time and find that the "complexities" correspond to regions of large λ_1 which peaks when they are about to change and that tend to disappear during evolution. The distribution of the intensity of these regions could be a measure of complexity. The rationale for this is that regions that would change a lot during a self-organization process do so because the are relatively complex.

Kolmogorov-Sinai entropy is traditionally calculated using Shannon entropy, but other entropy measures like Tsallis or Rényi entropy can also be employed [Sha06]. Different entropy measures have generally different meanings as well. As discussed in [BS95; Rue89], for example, KS-Rényi entropies with different parameter are related to the fractal dimension of the attractor.

Estimating KS-Entropy One way to estimate the entropy rate H of a random process $X = \{X_i\}$ with values in a finite alphabet $\mathcal{A}$ is to use an algorithm like the one proposed by Kontoyiannis [Kon+98]. With, for $i < j$, X_i^j denoting $\{X_i, X_{i+1} \ldots, X_j\}$ the algorithm calculates L_n, the minimum length k such that the sequence X_0^{k-1} that starts at time zero does not appear as a subsequence within the past X_{-n}^{-1}. Wyner and Ziv showed that L_n grows like $(\log n)/H$ in probability [WZ89], a result later refined to pointwise convergence by Ornstein and Weiss [OW93]. To account for the dependency on the starting position, an average over different starting positions is taken. Let $\Lambda_i^n(X) = L_n(T^i X)$, where $T^i X$ is the translated sequence. It has been shown that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \frac{\Lambda_i^n}{\log n} = \frac{1}{H},$$

almost surely and in L^1, under a condition known as the "Doeblin Condition" [Kon+98]. This condition holds for i.i.d. processes, ergodic Markov chains, and certain non-Markov processes. In a nutshell, the algorithm takes the mean of the longest run without repeating the past, divided by the logarithm of the length of the past, and then lets the past extend to negative infinity. A generalization of this method exists for random fields, as discussed in [Kon+98].

2.5.2 Approximate Entropy

A practical, computable alternative to Kolmogorov-Sinai entropy is *Approximate Entropy*, which is its lower bound [DM19; GP83].

To calculate the Approximate Entropy, of a time series of N equally spaced data points $u(i)$, given a window length m and a filtering level r, we proceed as follows: one constructs $n = N-(m-1)$ vectors of length m (m-grams), denoted as $x(k) = (u(k), u(k+1), \ldots, u(k+m-1))$. The $d(i,j) = \|x(i) - x(j)\|_\infty$ are then computed, and one counts the number of times the norm changes by less than r, taking its logarithm, and then computing the mean. This quantity scales with the

Lyapunov exponents of the system. The Approximate Entropy is obtained by computing the difference between this quantity for windows m and $m+1$. This provides a computationally feasible way to estimate a lower bound for the KS entropy, particularly for the case where the Rényi entropy form of KS entropy is considered with $q = 2$.

2.5.3 Permutation entropy

Permutation entropy was introduced in [BP02] as measure of complexity of time series. Unlike other measures, it is simple to calculate and robust to the introduction of noise. The idea is simply to calculate the entropy of the permutation patterns (called ordinal patterns in the literature [Ley+22]): using a sliding window of size n, we tally up the permutations that make the time series values increasing. Dividing by the total number of observations, we get the probability of observing a certain permutation during the dynamics. Specifically, we have the following definitions:

Definition 10 (Permutation entropy)**.** *let $\{x(t)\}_{t=0\dots T}$ be a real-valued time series and π_k^n the permutation that places the n elements starting at position k in increasing order. For example, for a time series $\{1, 2, 5, 3, -9, 4, \dots\}$, we have $\pi_0^3 = (1, 2, 3)$ and $\pi_1^3 = (1, 3, 2)$.*

Then the permutation entropy of order $n \geq 2$ is defined as the Shannon entropy of the distribution of permutations over the dynamics:

$$H_{perm}(n) := -\sum_{\pi \in \Pi(n)} p(\pi) \log p(\pi) \tag{2.5.4}$$

where the $p(\pi)$ are the relative frequencies of each of the $n!$ permutations

$$p_(\pi) = \frac{\sum_{k=0}^{T-n} \mathbf{1}\pi_k^n = \pi}{T - n} \tag{2.5.5}$$

Definition 11. *In the conditions of Definition 10, we define the permutation entropy rate per symbol of order $n \geq 2$ as*

$$h_n := \frac{H_{perm}(n)}{n - 1} \tag{2.5.6}$$

where division by $n - 1$ comes from the fact that comparisons start at the second value.

A key result is that Kolmogorov-Sinai entropy can be estimated from the permutation entropy in one dimension [AKK05; BKP02] under the assumption of ergodicity, and indeed in more general settings by complementing it with a measure of dispersion of trajectories characterized by the same permutation.

Multidimensional extensions of permutation entropy The extension of permutation entropy to the multidimensional setting is difficult. Whereas in one dimension the order in the permutation is the natural order induced by that of the underlying field, $\mathbb{R}$), in higher dimensions there is no natural definition of order. The simplest choice is to simply count the probabilities of each permutation across all different dimensions [Ley+22]. An extension of this approach, *weighted permutation entropy,* is to weight the probability of each permutation by their amplitude variation across dimensions [Fad+13].

The idea of including amplitude information, now in the one-dimensional setting, was also used in [Sun+14], where the authors construct states composed of permutation, amplitude pairs. Rather than encoding information in probabilities of observing each pair, the authors encode *transition* probabilities in the form of a graph, where each node represents a amplitude, permutation pair, with the connection weight between states representing the probability of transition between these states. Instead of entropy, as a measure of complexity, the authors use other graph-specific measures.

Again in the multidimensional setting but no longer in the time series context, in [Rib+12] the authors map images to the complexity-entropy plane defined in [Ros+07]. Essentially, the authors *construct* a time series from an image by sliding a square window of side L across it and simply apply the measure in [BP02]. Note that while the states (squares) built in such a way are unique and do not depend on the path, it is not clear what is the significance of a permutation of a flattened 2D array. The idea is simple, however, and was applied to images to plot entropy complexity plots of paintings [SPR18].

Finally, there are extensions using the using the minimum Rényi entropy [ZOR15] rather than the Shannon entropy to compute the complexity of the ordinal patterns.

2.6 Complexities of SEM images

In what follows we are interested in computing complexity measures of certain two-dimensional real fields fields $u(x, y) := u(\mathbf{x})$ on a square domain. These fields were produced on the surface of monocrystalline Nickel 001[8] by femtosecond laser irradiation, as described in detail in Section 4.5.1, and then recorded as SEM images. As mentioned on Section 2.1, we shall be interested in computing complexity measures of various transforms of such fields $f(u)$, with a particular emphasis on taking into account local or process information. The main goal of this section is to illustrate measures of complexity, and to compare and contrast them. We shall examine at least one measure for each of the axes proposed in Sec. 2.2.

Without loss of generality, throughout our discussion, unless otherwise stated image resolution, in pixels (512^2), as well as the number of gray levels (256) of the SEM images, are fixed.

2.6.1 Some notions of SEM image acquisition

It will now prove useful to discuss some of the details of Scanning electron microscopy, in order to understand the relationship between observed images and underlying fields. An in-depth treatment is outside the scope of this thesis, but the interested reader may wish to consult [Rei00] or [Gol+18a].

In Scanning electron microscopy, a sample of material is exposed to an electron probe emitted from an electron gun. These electrons are absorbed by the material and interact within a drop-shaped region called the *interaction volume*. The details of the interaction between matter and electrons is complex, and depends, amongst other things, on material and topography. This complex process results in the emission of (amongst others) electrons called *secondary electrons*, which are detected by a current detector placed at some distance above the sample.

The secondary electron current emitted from the sample is proportional to the intersection between sample and interaction volume, which changes with topography. The energy of the beam also plays a role (see e.g. Fig.[Gol+18b] or Fig. 5 in [Baa+21]).

[8] The numbers are Miller notation, and indicate that the sample is composed of monocrystalline Nickel, which has face-centered crystal structure, and that the sample is parallel to one of the faces of cube.

The signal at the detector at the position of the beam is amplified and converted into a pixel intensity value, which for the sake of simplicity, we shall assume can take one of 256 values. The collection of pixel intensities at a grid thus creates an array which is converted to an image file.

This image maps to the geometry of the sample. At the edges, a greater number of electrons can be emitted since a significant portion of the interaction region is in close proximity to the surface. At the trenches, fewer secondary electrons may escape and make their way to the detector, as they may be obstructed along their path. Consequently, the detector behaves like a virtual light source, making objects with a geometry that is oriented towards the detector appear brighter.

See Fig. 2.2, reproduced from [Sch20], for a schematic representation illustrating this process.

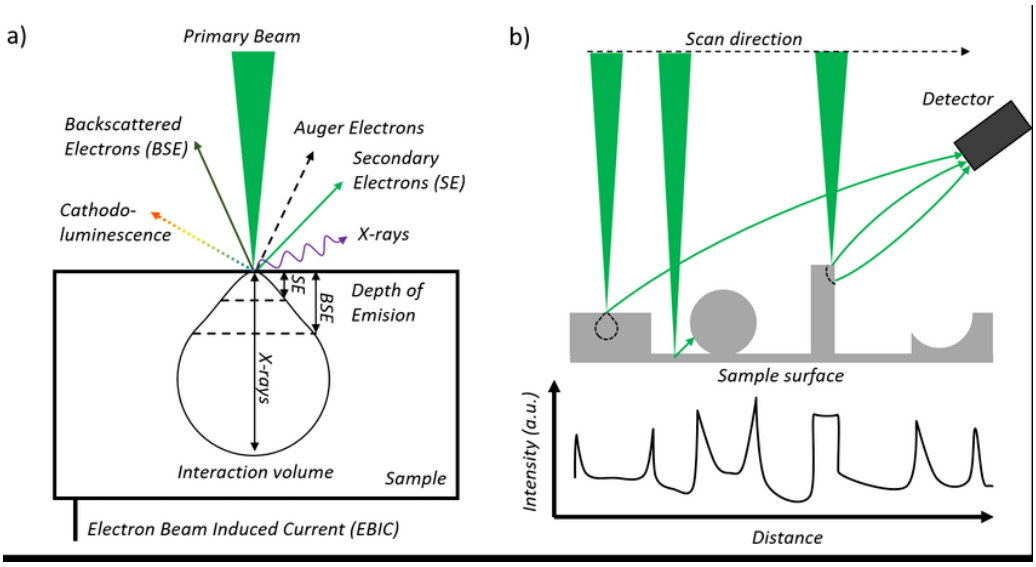


Figure 2.2: Reproduced from [Ara+19], Fig. 21: Interaction volume and contrast mechanism in SEM. a) Schematic view of the interaction volume of a focused electron beam with a bulk sample. Different signals are generated which can escape from different depths below the sample surface. b) Contrast generation in SEM: at different points of a sample geometry different intensities of SE are detected with edges being bright. If a nanostructure has a comparable size to the interaction volume it will appear bright as a whole.

It is possible to go beyond this simple relationship and recover actual height information from SEM images. Unfortunately, most methods to do so crucially depend on multiple image views[Taf+15], which were unavailable to us. Image height reconstruction can also be achieved for single images, via sophisticated machine learning models[Hou+22] or expert models[Ara+19] relying on pre-generated shape databases and Atomic Force Microscopy (AFM) images[BQG86], which do contain height information.

Most of these models being inaccessible to us, and the accuracy of commercial packages being disputed[TVV17], we instead make the simple approximation that the intensity of the image is proportional to the amplitude of the field u. This is true to first order for sufficiently high-energy beams compared to height of the features this approximation holds, as the intersection between the

interaction volume and the surface increases with height (see [Ara+19], Fig.2). It is the basis for the height estimation method in [Ara+19], which provides with good results.

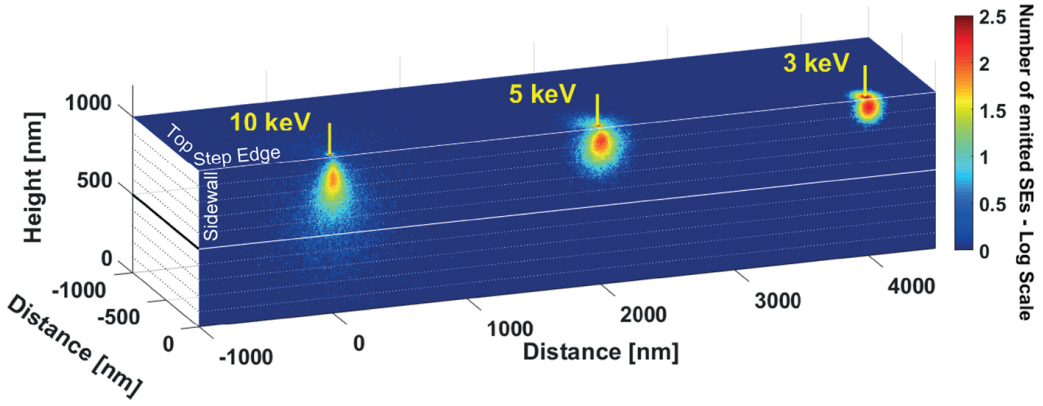


Figure 2.3: Legend reproduced verbatim from [Ara+19], Fig. 1: SE emission sites on the top and the sidewall of a (1 µm high) step are shown for three acceleration voltages (10, 5, and 3 keV). The zero-diameter electron beam (with 104 electrons) lands on top of the step, 50 nm away from the edge. The white dashed lines on the sidewall aid in judging how many electrons can actually be emitted from a particular depth of the step.

2.6.2 Estimating field amplitude distribution from SEM images

The SEM images available to us were acquired during several experimental sessions. The details of the image processing vary with session and laser parameters, but typically the highest gray level value does not correspond to the highest field value, rather to the threshold of the detector, which saturates at a certain level. SEM images are typically optimized for visualization and hence processed to capture the maximum possible dynamic range, by (automatically) setting the maximum and minimum of each image so that most pixels in the image are represented in the middle range. This make images more aesthetically appealing, but it also randomly modifies the scale of each image: image gray levels for two different images typically correspond to different detector intensities. It also saturates the bottom level in the same way as the top level discussed above.

This transformation will of course impact complexity measures which depend on the gray level histograms. To see how, consider Shannon entropy, which as discussed above, given state of knowledge encoded in the form of a probability distribution, is the mean surprise (a decreasing function of probability) when observing events which follow that distribution.

To encode knowledge about the amplitude of u, we take its representation as a 8-bit image I and then build and subsequently normalizes a histogram binning each of the 256 possible pixel intensity values. We thus obtain a probability mass function p_i representing the probability that one finds the gray level i among the image pixels. Crucially, assuming that image intensity is proportional to field amplitude, this probability mass function is that of the field amplitudes as well. The complexity of the field amplitudes can then be characterized by the Shannon entropy of

the gray levels of the image, which is a proxy for the entropy of the field:

$$H(I) = -\sum_{i=0}^{255} p_i \log p_i.$$

We note about the entropy of the image gray levels $H(I)$

1. It is roughly independent of image size
2. It does not take into account local information
3. It is very dependent on the entropy of the saturation

Maxent distribution of the saturated levels Given an 8-bit image (hence 256 possible gray levels) of size N, assume that there are $n < N$ pixels at the saturated level. If n is large, then the entropy of the image is low, as the saturated level is a good gray level guess: the image is unsurprising. But this is an artefact of image acquisition, and cannot correspond to the entropy of the physical field, for which the intensity is not bounded above by a threshold.

To estimate the entropy of the physical field, we must model the distribution of gray levels above the threshold. A simple approach is to assume a fixed number of possible gray levels above the threshold (256 for simplicity). According to the principle of parsimony, assume that each is equally likely; then according to the coarse-graining property, the entropy of the field is upper bounded by $(1-p_{255})H(I)+p_{255}(\log n)$, where the term in parenthesis is the entropy of the uniform distribution. Note that the greater p_{255}, the more the entropy of the saturated part will be important. Also, note that even if p_{255} is small, there is always a large enough n that will make the second term dominate. A more sophisticated model, which we use in this report, is to take an exponential distribution (which is the maxent distribution with support on the positive reals given mean), truncated so that the resulting image has 512 gray levels.

$$p(x|\lambda, b) = \frac{\frac{1}{\lambda}e^{-\frac{x}{\lambda}}}{1 - e^{-\frac{b}{\lambda}}},$$

where $\lambda > 0$ is the exponential distribution scale parameter and $b > 0$ is the threshold (that is $0 < x \leq b$). See Fig. 2.4 for a visualization of the results.

All images in this Chapter are thus 9 bits where the degenerate level was resampled from a shifted truncated exponential distribution, the scale parameter of which is inferred by continuity (see supplementary material/code).

2.6.3 Complexities of SEM images

In this section, we compute, visualize, and compare several entropies and complexities of SEM images. Since entropies are measures of complexity associated to representations of states of knowledge given in terms of a probability distribution, we shall endeavour to specify what knowledge is being encoded. The SEM images that constitute the object of this analysis can often be described in terms of *patterns*. We shall therefore focus on the distributions that take into account this notion: of gray level runs, of Fourier modes, of particular-size patches of the image, for example. After settling on the encoded knowledge, there remains the actual measure of complexity associated with this distribution. We thus present, for each distribution, the Shannon, Rényi, and Tsallis entropies, and the associated complexities as defined in Sec. 2.4.

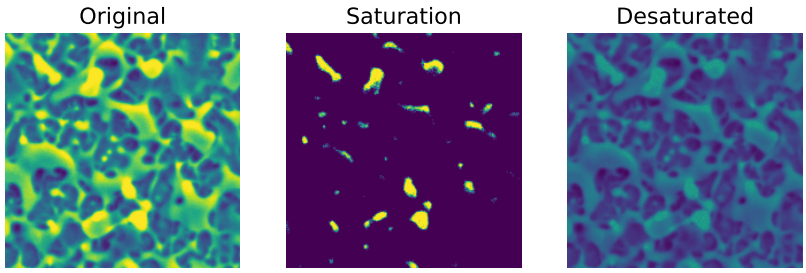


Figure 2.4: SEM images (left) may be saturated at the top gray level. The percentage of saturated pixels can be very high (middle), which will strongly impact image entropy. We remove saturation by resampling saturated pixels from a shifted truncated exponential distribution (right)

Of gray levels

We start our illustration of measures of complexity applied to SEM images with the *gray level distribution* of an image, which consists in the histogram of gray levels (typically 0-255) of the image. As this histogram is created simply by counting how many pixels have a certain gray value, the histogram loses all spatial and contextual/local information. For example, two entirely different images can have exactly the same histogram but differ in texture, edge orientation, and object arrangement. Consider e.g. a checkerboard and a zebra stripe pattern; both may have an equal number of black and white pixels, yielding identical gray-level distributions. However, the spatial relationships between the pixels, which define the actual patterns, are entirely different. In the case of SEM images in Fig. 2.5, we observe that, for example, the "nothing" image and the "faint" image have similar distributions.

Another interesting thing to observe is the dependency of the Tsallis and Rényi entropies and complexities on the respective parameters, in Figs. 2.8, 2.9, and 2.14, 2.7, respectively. Changing the magnitude of the parameters changes the values of the entropies, but it does not change the *rankings* of the SEM images with respect to this measure. However, changing the parameter does affect the ranking in the case of the complexity measures. This opens the door to principled parameter selection, which would be set in such a way that the ranking of complexities of baseline fields matches intuition/application.

As another illustration, we apply these measures to random samples of five series of SEM images obtained at constant laser fluence and time delay, and varying the N parameter, as depicted in Fig. 2.19, obtaining Fig. 2.22, As explained in Sec.3.2.4, these series can be roughly regarded as depicting a temporal evolution (in samples from a time-varying distribution sense). Generally speaking, entropies of gray levels tend to increase with N, and complexities to decrease.

Finally, we apply the several measures of complexity to the fluence-delay plane presented in [Nak+22], in Section A.1.

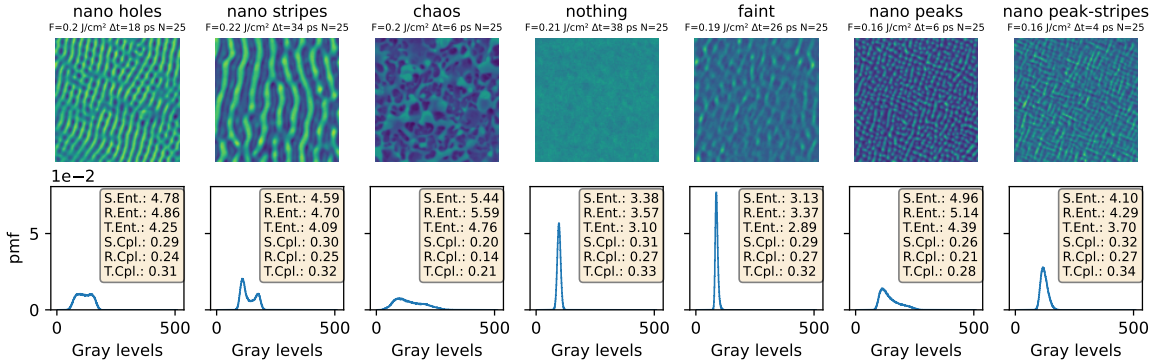


Figure 2.5: Measures of complexity of gray levels (distributions in blue, bottom row) of selected SEM field images (top row): Shannon, Rényi, Tsallis entropies and complexities (inset, bottom row; entropies in bits and complexities in bits^2). Gray levels above 256 are inferred using the strategy described in 2.6.2 . We set $q = 1.05$ and $\alpha = 0.5$ in Tsallis and Rényi measures, respectively to stay within the same order of magnitude.

Of gray level runs

Another way to include local information in the probability mass function is to consider the gray level runs [Gal75], a feature transformation that was originally proposed in the context of visual texture analysis. A gray level run is a set of consecutive, pixels having the same gray level along some chosen direction. The authors in [Gal75] propose a number of measures, but here we shall define an entropy for consistency: after building the gray level run matrix (GLRM), we can simply consider the Shannon entropy of the gray level run (GLR) matrix "image". With g denoting pixel intensity, the GLRM collects the unnormalized probabilities of runs of length r, $p_\theta(r|g)$ along a given direction θ (angle, in degrees, with respect to the original image orientation). The entropy of the GLR "image" is thus $H(R|G)$, and we denote it GLRE1.

Another possibility, since the marginal probability $p_\theta(g)$ can be inferred from the image gray levels histogram, is to compute the entropy of the joint probability mass function $p_\theta(r, g) = p_\theta(r|g)p(g)$. Assuming a square image of L pixels side, we define GLRE2 as

$$H_\theta(r, g) = -\sum_{g=0}^{255}\sum_{r=1}^{L} p_\theta(r, g) \log p_\theta(r, g) \tag{2.6.1}$$

GLRE dependence on direction By construction, the GLRM depends on the direction. As can be seen in Figs. 2.11c, 2.11a, 2.11b, GLRE1 depends strongly on image orientation, in particular for anisotropic images, as can be observed in Figs. 2.11a, 2.11b. On the other hand, the GLRE1 of isotropic images is roughly constant with respect to the GLRM direction, as can be seen in figure 2.11c.

To account for this variability, we simply pick the direction that minimizes the direction-dependent GLRE1 2.6.2, because the orientation of the images is arbitrary (thus has no physical significance) and because the intuitive notion of pattern complexity does not depend on orientation

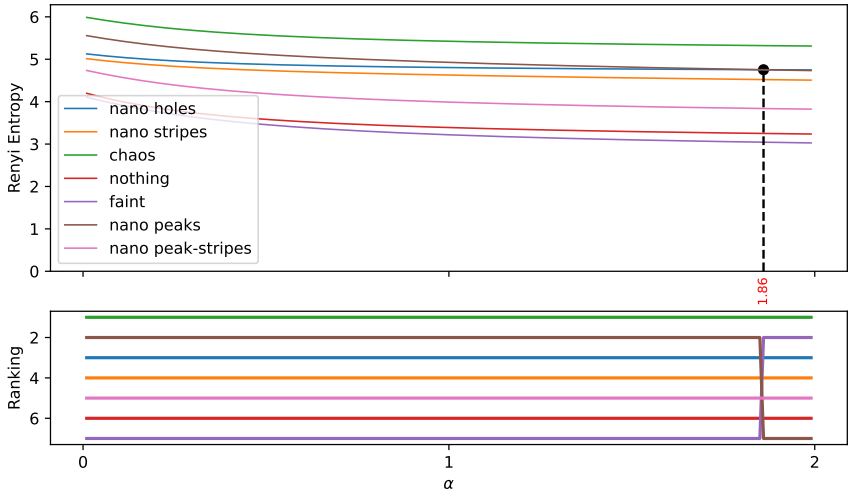


Figure 2.6: Comparative analysis of Rényi entropy **of the gray levels** across various laser-induced patterns. The top plot shows the Rényi entropy as a function of the parameter α for seven different patterns. Intersections between the curves are marked with vertical lines and annotated. The bottom plot ranks the patterns based on their entropy values, with the pattern having the highest entropy ranked as 1. As explained in Section 2.3.5, the interpretation of Rényi entropy changes for different values of α. Interestingly, the *ranking* of the SEM images with respect to this measure stays roughly constant.

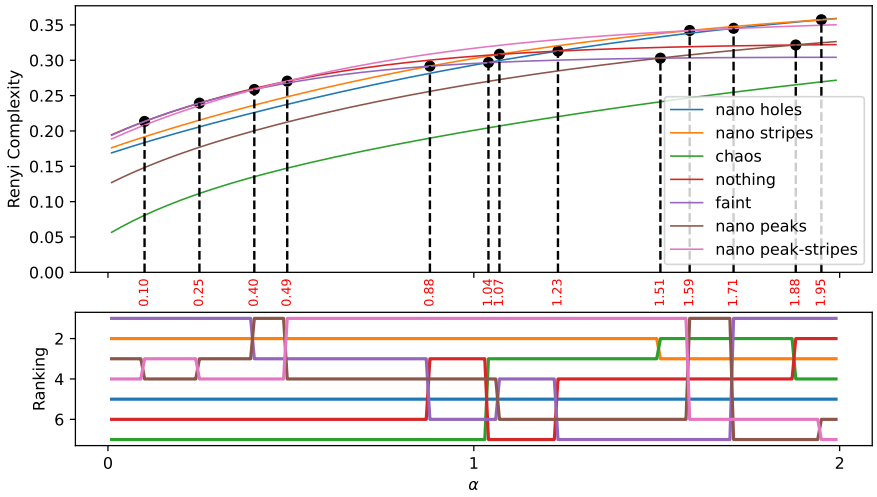


Figure 2.7: Comparative analysis of Rényi complexity **of the gray levels** across various laser-induced patterns. The top plot shows the Rényi complexity as a function of the parameter α for seven different patterns. Intersections between the curves are marked with vertical lines and annotated. The bottom plot ranks the patterns based on their complexity values, with the pattern having the highest complexity ranked as 1. As explained in Section 2.3.5, the interpretation of Rényi entropy changes for different values of α. Interestingly, whereas for Rényi entropy, the ranking of the SEM images stayed constant for different values of the parameter, for Rényi *complexity* the different interpretations have a dramatic impact in the ranking of the SEM images.

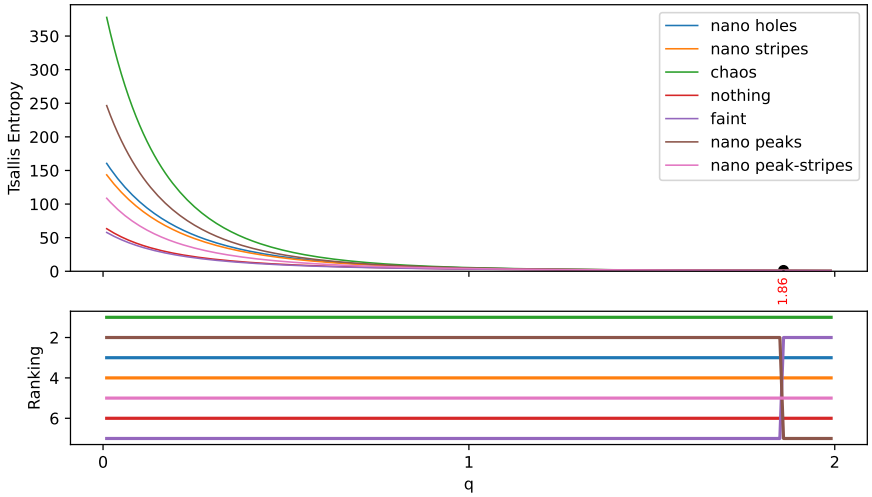


Figure 2.8: Comparative analysis of Tsallis entropy **of the gray levels** across various laser-induced patterns. The top plot shows the Tsallis entropy as a function of the parameter q for seven different patterns. Intersections between the curves are marked with vertical lines and annotated. The bottom plot ranks the patterns based on their entropy values, with the pattern having the highest entropy ranked as 1. As explained in Section 2.3.4, the interpretation of Tsallis entropy changes for different values of q. Interestingly, the *ranking* of the SEM images with respect to this measure stays roughly constant.

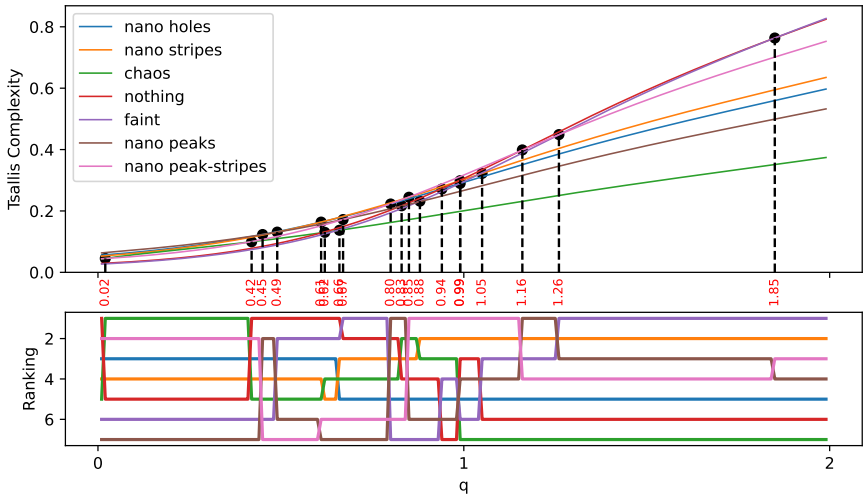


Figure 2.9: Comparative analysis of Tsallis complexity **of the gray levels** across various laser-induced patterns. The top plot shows the Tsallis complexity as a function of the parameter q for seven different patterns. Intersections between the curves are marked with vertical lines and annotated. The bottom plot ranks the patterns based on their complexity values, with the pattern having the highest complexity ranked as 1. As explained in Section 2.3.4, the interpretation of Tsallis entropy changes for different values of α. Interestingly, whereas for Tsallis entropy, the ranking of the SEM images stayed constant for different values of the parameter, for Tsallis *complexity* the different interpretations have a dramatic impact in the ranking of the SEM images.

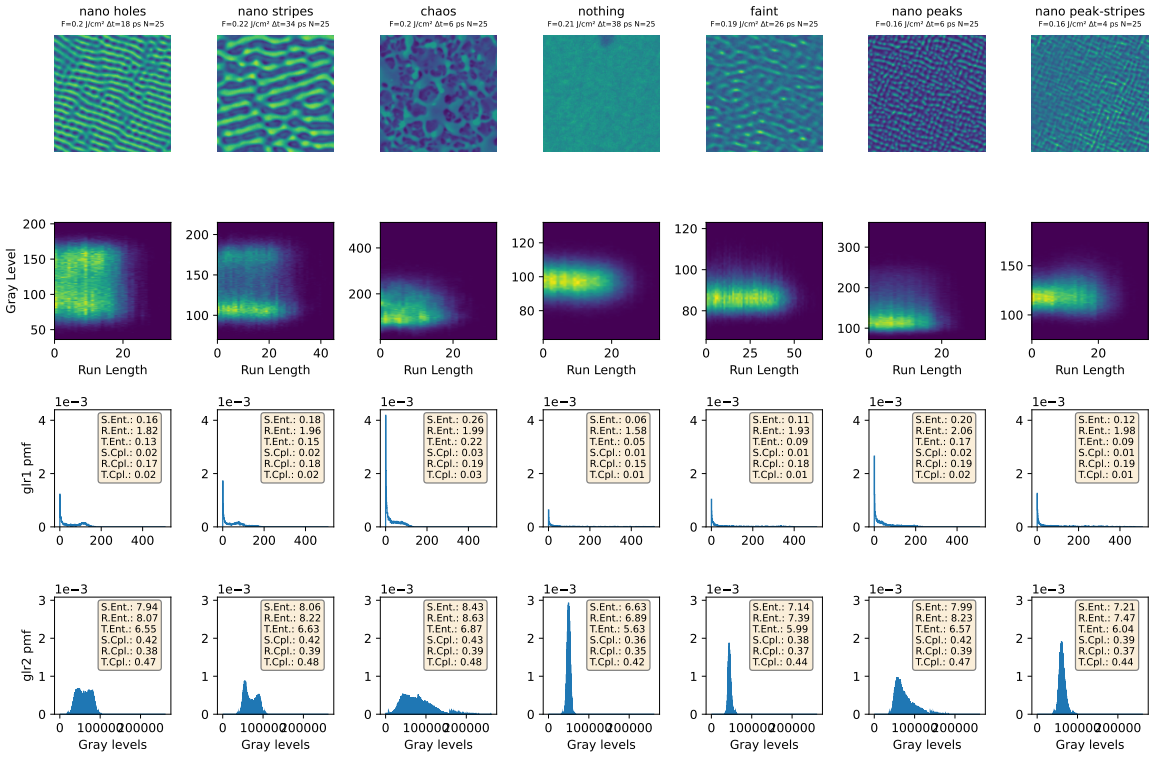
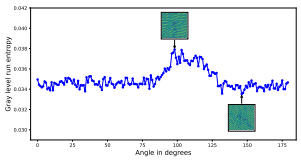
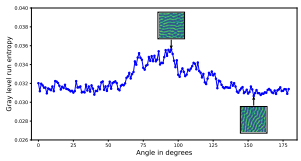


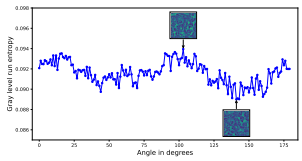
Figure 2.10: Measures of complexity of gray level runs (glr1 and glr2) (distributions in blue, bottom rows; for the glr1 distribution, we shown only non-zero run lengths) of selected SEM field images (top row): Shannon, Rényi, Tsallis entropies and complexities (inset, bottom row; entropies in bits and complexities in bits^2). Gray level runs matrices are depicted on the second row from the top (note the different ranges). Gray levels above 256 are inferred using the strategy described in 2.6.2 . We set $q = 1.05$ and $\alpha = 0.5$ in Tsallis and Rényi measures, respectively to stay within the same order of magnitude.



(a) GLRE1 as a function of angle for nanoholes.



(b) GLRE1 as a function of angle for stripes.



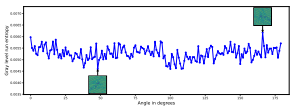
(c) GLRE1 as a function of angle for chaos.

Figure 2.11: Comparison of GLRE1 as a function of angle for different types of images.

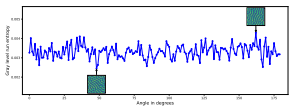
either, we describe a particular pattern as "ordered" if there is one or several particular directions along which it appears to be "ordered". Precisely:

$$H_{GLR}(r,g) = \min_{\theta} \{H_\theta(r,g)\} \tag{2.6.2}$$

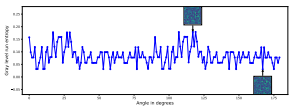
A GLRE1 heatmaps of entropy and complexities applied to an experimental range of SEM images can be observed in Section A.1.2. GLRE2, on the other hand, is much less dependent on orientation than GLRE1, as can be seen in Figs. 2.12c, 2.12a, 2.12b, which justifies taking a single random orientation in GLRE2 computations. GLRE2 heatmaps for several measures of complexity applied to an experimental range of SEM images can also be observed in Section. A.1.2.



(a) GLRE2 as a function of angle for nanoholes.



(b) GLRE2 as a function of angle for stripes.



(c) GLRE2 as a function of angle for chaos.

Figure 2.12: Comparison of GLRE2 as a function of angle for different types of images.

Of the Fourier spectrum

Pattern formation in self-organization systems is commonly described in terms of spacial frequency selection [CH93], since in the frequency domain, a dominant spacial frequency corresponds to a wavelength that occurs more frequently. Simple patterns in the frequency domain, having low Shannon entropy, will have a small number of frequent spacial frequencies. A pattern with just one spacial frequency has a Dirac delta power spectrum.

In the same way that there is no natural way to take into account spacial information in the binning in the time domain, also in the frequency domain there is a degree of arbitrariness on how to build the 2D histogram.

A simple procedure is to average 1D spectra in all directions around the center building an azymuthal power spectrum. Once this one-dimensional power-spectrum is obtained, we can use the approach in [KK92]: given a field $u : \mathbb{R} \to \mathbb{R}$ with discrete Fourier transform $\mathcal{F}[u(x)] := \hat{u}(k)$, we

define its power Spectral density distribution

$$p_i := \frac{|\hat{u}_i|^2}{\sum_i |\hat{u}_i|^2}$$

and the *spectral entropy* as the Shannon entropy of this distribution

$$H_1^{spec} = -\sum_i p_i \log p_i \tag{2.6.3}$$

As explained above, H_1^{spec} measures the average surprise in the distribution of powers of different spacial frequencies, *for a given choice of averaging procedure*. For the azymuthal averaging above, the entropy measures average surprise in spacial frequency independently of orientation (see Fig.. 2.13, fourth row).

This idea can be generalized straightforwardly to the two-dimensional case. Given $u : \mathbb{R}^2 \to \mathbb{R}$ with discrete Fourier transform $\mathcal{F}[u(\mathbf{x})] := \hat{u}(\mathbf{k})$, we define its two-dimensional power Spectral density distribution

$$p_{ij} := \frac{|\hat{u}_{ij}|^2}{\sum_{ij} |\hat{u}_{ij}|^2}$$

and the *spectral entropy* as the Shannon entropy of this distribution

$$H_2^{spec} = -\sum_{i,j} p_{ij} \log p_{ij} \tag{2.6.4}$$

The entropy measures now the average surprise in the distribution of powers of different spacial frequencies, where direction is taken into account (see Fig. 2.13, third row). Similarly, we examine the complexities of the distribution of the phase part of the Fourier transform, which is known to hold structural information (see Fig. 2.13). Intuitively, if one perturbs the phase spectrum, the modes will interfere in unexpected ways, with an overall effect that will tend to destroy original image features.

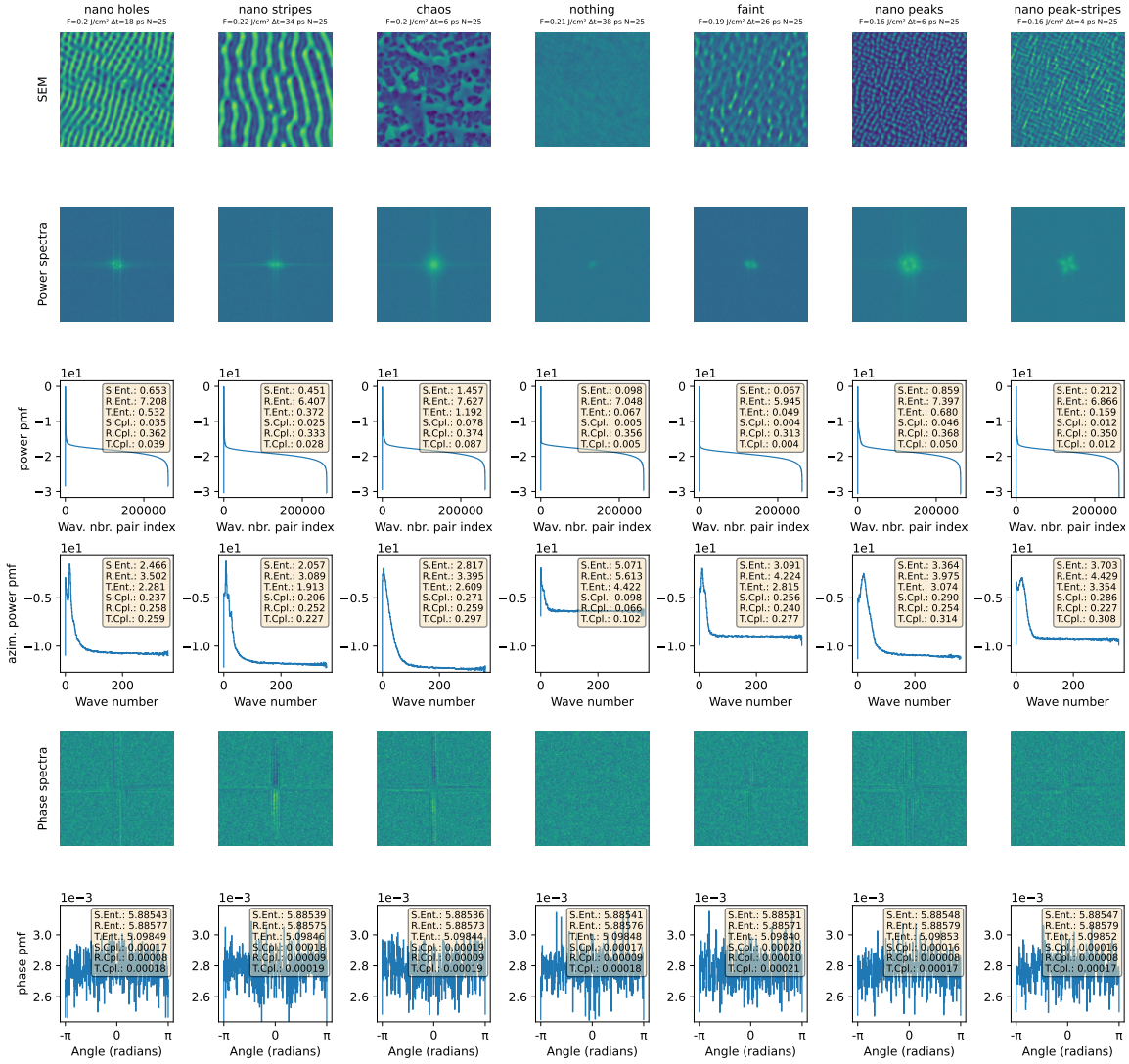


Figure 2.13: Complexity and entropy measures of spectra of selected SEM field images. The top row displays the SEM images, each labeled with its corresponding laser fluence, pulse duration, and number of pulses. The second row shows the (log) power spectra of these fields. The third row presents the sorted power spectrum probability mass function (pmf), log transformed for ease of visualization, with inset boxes indicating Shannon, Rényi, and Tsallis entropies and complexities. The fourth row displays the azimuthal power spectrum pmf, also log transformed for ease of visualization, with inset boxes for entropies and complexities. The fifth row shows the phase spectra, and the bottom row presents the phase spectrum pmf, again with inset boxes for entropies and complexities. All pmfs are computed using the strategy described in 2.6.3. All entropies are measured in bits and complexities in bits^2. We set $q = 1.05$ and $\alpha = 0.5$ for Tsallis and Rényi measures, respectively, to maintain a consistent scale. . Note the phase spectral distributions are almost uniform, which explains the low variance on complexity measures.

2.6.4 Self-correlation

In this section, we study the internal degree of organization of the SEM images, which places us squarely in axis 3. The idea is to measure the degree of organization of an SEM image by the measuring self-similarity.

We do this in two different ways: (i) we measure the similarity between different same-size patches of the image. It is clear that an image for which composing parts are either featureless or display a high degree of organization, with the same pattern repeatedly displayed, will score high in similarity. As in previous measures, there is the matter of scale which will strongly impact the performance of this measure. And (ii) we measure the similarity between different *scales* of the same image, by comparing different scaled versions of patches of the same image. An image that is scale invariant will score high with respect to this measure. We propose a third measure of similarity which is simply one of the Pythagorean means of the other two.

For simplicity and ease of implementation, we use cross correlation as a similarity measure, but other similarity measures could arguably be used as well. Cross correlation has been used since the 1970s [KWS75; Anu70], and is a standard measure of image similarity in Digital Image Correlation[SOS09]. Given fields $u, v \in \mathbb{R}^2$, we define their *cross-correlation* $c(u, v) \in \mathbb{C}^n$ as the inverse Fourier transform of the Hadamard product of the transform of u and the conjugate of the transform of v:

$$c(u, v) = \mathcal{F}^{-1}\left\{\mathcal{F}\{u\} \circ \mathcal{F}\{v\}^*\right\} \tag{2.6.5}$$

where the circle denotes the elementwise Hadamard product and $\mathcal{F}, \mathcal{F}^{-1}$ denote, respectively, the Fourier transform and the inverse Fourier transform. We normalize this quantity by taking

$$C(u, v) = \frac{c(u, v)}{\sqrt{c(u, u) \times c(v, v)}}$$

Finally, this allows us to define *correlation strength* $r(u, v)$, which will be used as a measure of image similarity.

Definition 12 (Correlation strength)**.** *In the conditions and notation above, we define correlation strength between real two-dimensional fields u and v as the maximum*[9] *of the absolute value over the domain of $C(u, v)$. If the common domain of u and v is discretized as an $n \times n$ array (meaning $C(u, v)$ is represented by a $n \times n$ complex matrix), we have*

$$r(u, v) = \max_{i,j=1\cdots n} |C(u, v)_{ij}| \tag{2.6.6}$$

We use the Fast Fourier Transform [CT65] in our implementation of this measure, which makes computation expedient.

Cross-patch similarity We define the *cross-patch similarity* of a field $u \in \mathbb{R}^n$ at a scale s as the mean correlation strength for field patches of size $s \times s$. With S denoting all the s by s patches of

[9] Other norms other than the max norm are used in the literature [SOS09] but they are less robust to noise and spurious correlations. By looking at the maximum peak, we are less likely to be affected by noise because it is unlikely for noise to cause a very high peak at the wrong alignment.

u, we have

$$\mathrm{CPS}(u) = \frac{1}{|S^2|} \sum_{v,w \in S} r(v, w) \tag{2.6.7}$$

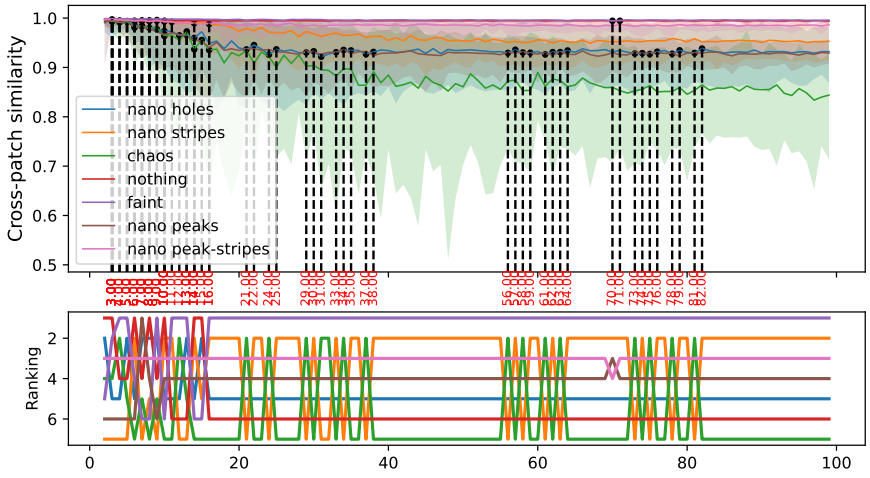


Figure 2.14: Comparative analysis of mean cross-patch similarity across various laser-induced patterns. The top plot shows the mean cross-patch similarity, with min-max error bands, as a function of the square patch side for seven different patterns. Intersections between the curves are marked with vertical lines and annotated. The bottom plot ranks the patterns based on their mean cross-patch similarity values, with the pattern having the highest similarity ranked as 1. As can be seen on that plot, after an initial stage (up to 20-patch side) of some variation (for nano-stripes and chaos in particular), the ranking remains constant, which is in agreement with the existence of local features of different scales for some of the patterns.

Cross-scale similarity We define the *cross-scale similarity* of a field $u \in \mathbb{R}^n$ at scale s as the mean correlation strength for field patches across scales at that scale. To compares patches at different scales, one scales them up or down until they are at the desired scale t. For simplicity, we shall only consider square patches with an even number of pixels. With T now denoting all the $t = 2n$ by $t = 2n$ patches of u that can be obtained by scaling up or down other even-side patches of u, we have

$$\mathrm{CSS}(u) = \frac{1}{|T^2|} \sum_{v,w \in T} r(v, w) \tag{2.6.8}$$

Internal similarity The measures defined above see "organization" in different ways, either as organization as similarity between different parts or organization as similarity between different

scales. In order to create a combined measure, we study three alternatives. The first is the simply the mean of the two. Repeating Shannon's argument[Sha48], additivity is a desirable property of a natural measures of complexity, we can easily justify the first. To justify using a multiplicative notion of complexity, we appeal to the work described in Section 2.4, where complexity is defined as the product of a dissimilarity and an entropy. The harmonic mean is the third Pythagorean mean, and can be seen as a combination of the other two. We recall that we adopt Bennet's functional definition of complexity as "whatever increases when something self-organizes", which will allow us to assess the quality of a complexity measure by how well it conforms to observations.

Definition 13 (Internal similarity)**.** *In the notation and conditions above, we propose three measures of internal similarity of a real field in two dimensions u, respectively, arithmetic, geometric, and harmonic:*

$$\mathrm{IC}_a = \frac{1}{2}\left(\mathrm{CSS}(u) + \mathrm{CPS}(u)\right) \tag{2.6.9}$$

$$\mathrm{IC}_g = \sqrt{\mathrm{CSS}(u) \times \mathrm{CPS}(u)} \tag{2.6.10}$$

$$\mathrm{IC}_h = \frac{2}{\frac{1}{\mathrm{CSS}(u)} + \frac{1}{\mathrm{CPS}(u)}}, \tag{2.6.11}$$

2.6.5 Lempel-Ziv complexity

Kolmogorov complexity of a sequence, the size of the shortest program on a Universal computer that can compute it and halt, is not effectively computable. But arbitrarily restricting the encoding to use only recursive copy and recursive paste operations [ZRB05; LZ76], we obtain a complexity measure that is effectively computable and approximates Kolmogorov complexity asymptotically [Gra12].

To compute the Lempel-Ziv complexity of a sequence, do:

1. Break the sequence S into words $W_0 = \emptyset$, and W_{k+1} being the shortest new word immediately following W_k. For example, the sequence $S = 11010100111101001\ldots$ is broken into $(1)(10)(101)(0)(01)(11)(1010)(01\ldots)$.

2. Each word W_k, with $k > 0$, is an extension by one digit s_{last} of some other word W_j, with $j < k$. It is encoded simply by the pair (j, s_{last}). In the example above, we have $(0,1),(1,0),(2,1),(0,0),(4,1),(1,1),(3,0),\ldots$. Thus, the encoding is lossless, as it can be uniquely decoded.

3. Count the number of words. This is the Lempel-Ziv complexity.

To compute a complexity measure in the spirit of Section 2.4, we use *normalized compression distance* the similarity measure proposed in [Li+04], which is essentially a compression-based estimate of the relative Kolmogorov complexity between strings. With $x^n := x_1 x_2 \cdots x_n$ denoting a string composed of n symbols, and $x^n y^n$ denoting the concatenation of strings x^n and y^n, and $C_{LZ}(x^n)$ denoting an estimate of Kolmogorov complexity of the string x^n via some compression algorithm (in our case, Lempel-Ziv), we have

$$\mathrm{NCD}_{LZ}(x^n, y^n) = \frac{C_{LZ}(x^n y^n) - \min\{C_{LZ}(x^n), C_{LZ}(y^n)\}}{\max\{C_{LZ}(x^n), C_{LZ}(y^n)\}} \tag{2.6.12}$$

	SEM		Power Spectrum		Phase Spectrum	
	LZ	I-LZ	LZ	I-LZ	LZ	I-LZ
nano holes	0.15108	0.09414	0.94121	1.83606	0.96376	1.89159
nano stripes	0.14143	0.10517	0.94146	1.83730	0.96382	1.89185
chaos	0.16396	0.07624	0.94172	1.83841	0.96395	1.89282
nothing	0.12296	0.11949	0.94093	1.83557	0.96375	1.89114
faint	0.10722	0.12473	0.94228	1.83921	0.96384	1.89180
nano peaks	0.15751	0.08520	0.94174	1.83789	0.96387	1.89207
nano peak-stripes	0.14293	0.10362	0.94136	1.83703	0.96381	1.89327

Table 2.1: Lempel-Ziv (LZ) complexity and Intensive LZ Complexity (I-LZ) values for original 512 square pixel SEM field samples, their Power Spectrum, and their Phase Spectrum for different image samples. Note the low variance of the values for the Power and Phase Spectra.

Hence, the complexity measure that we propose is a compression based analog of the intensive statistical complexity (cf. eq. 2.4.2. We define *Intensive Lempel-Ziv complexity*:

$$C_{LZ}(x^n) = \alpha \frac{\mathrm{NCD}_{LZ}(x^n, x_*^n) C_{LZ}(x^n)}{C_{LZ}(x_*^n)}, \tag{2.6.13}$$

where we take x_*^n to be some sufficiently random sequence (which in our case consists of n samples from the uniform distribution $U(G)$, where G is the number of gray levels of the original image).

2.6.6 Taylor Entropy

The idea is simple: we see a large pattern u as a collection of independent long-time evolutions from slightly perturbed initial conditions and average boundary conditions via a local process f. Since there are no long-range interactions, the correlation between different patches u_i, u_j of the pattern u are the result of a composition of local interactions.

We can thus see each sub-patch of the same image as the result of perturbed initial conditions, and the diversity of such patches as a measure of sensitivity to initial conditions, in the same manner as KS-entropy.

To have a measure of diversity of patches u_i that takes into account spatial information, we look at all the derivatives that are present in f. Specifically, if $f(x_i, \ldots, \partial_i^j, \ldots)$ contains a certain number of spatial derivatives, we consider the smallest patch size k such that all the spatial derivatives in f can be computed using a finite difference scheme in k(we say that the derivatives are *compatible* with the patch size). The patch size defined in this way is a measure of locality of the process.

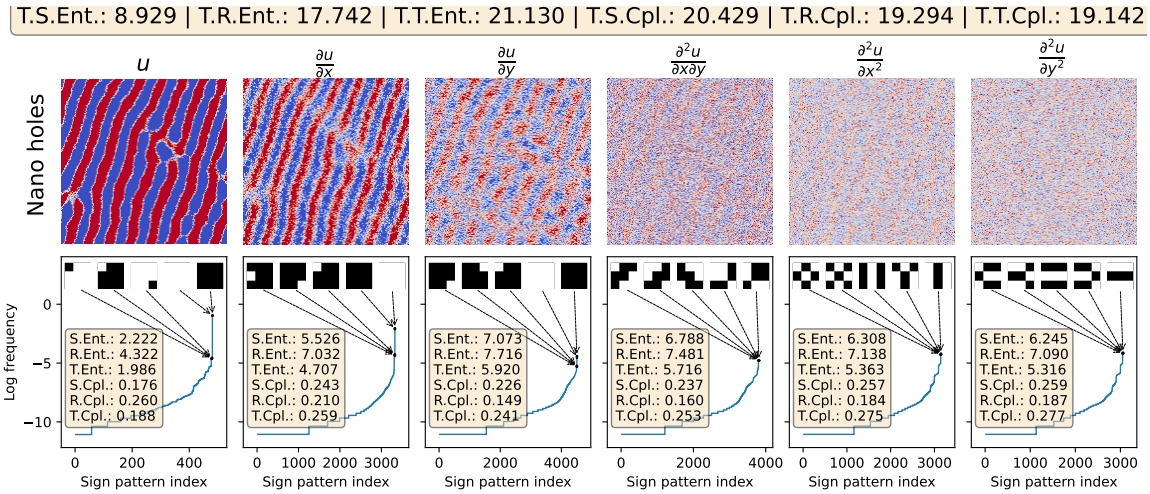
In this setup, there is no natural order between patches: it is meaningless to say that patch u_i comes before or after u_j, and so it is making as-hoc ordinal patterns.

But one does not need to. As each patch represents a different evolution of the system from perturbed initial conditions, the diversity of the patches determines the sensitivity to initial conditions, which is a measure of the complexity of f. For simplicity, we measure patch diversity using the distribution of the sign $\{-, 0, +\}$ patterns of the derivatives compatible with the patch size. For a 3 by 3 patch, which is compatible with derivatives of up to order two, for each derivative there

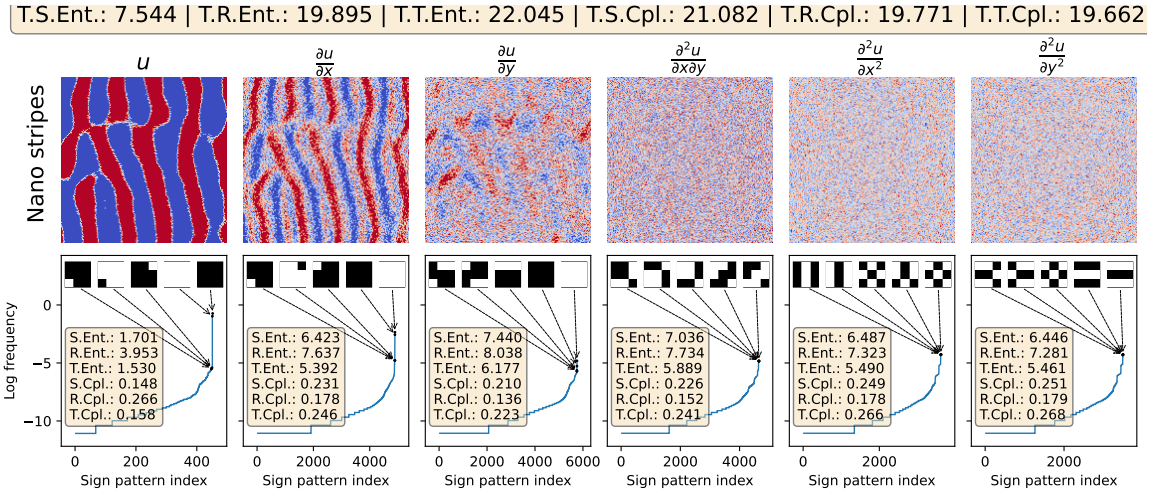
are 3^9 possible patterns. As each of the derivatives is a term in the Taylor expansion, they are independent (by Taylor's theorem). Hence, the diversity of the patterns can be obtained by adding the diversities of the individual terms, weighted by the coefficients of the Taylor expansion:

Definition 14. *Given a multidimensional field u, we define its* Taylor entropy *of order n as the sum of the entropies of the sign pattern distributions of all the spatial derivatives up to order n, weighted with Taylor series coefficients. Patch size is defined as the minimum patch size that allows calculating all derivatives up to order n using a finite difference stencil.*

This entropy can be calculated using Shannon, Rényi, or Tsallis measures of uncertainty, and complexity: cf. Figs. 2.15, 2.16, 2.17, and 2.18 for a visualization. It should be added, however, that the super- and sub- additivity 2.3.8 of Tsallis entropy implies that combining the derivative entropies needs to be done differently. This will be the subject of a future investigation.

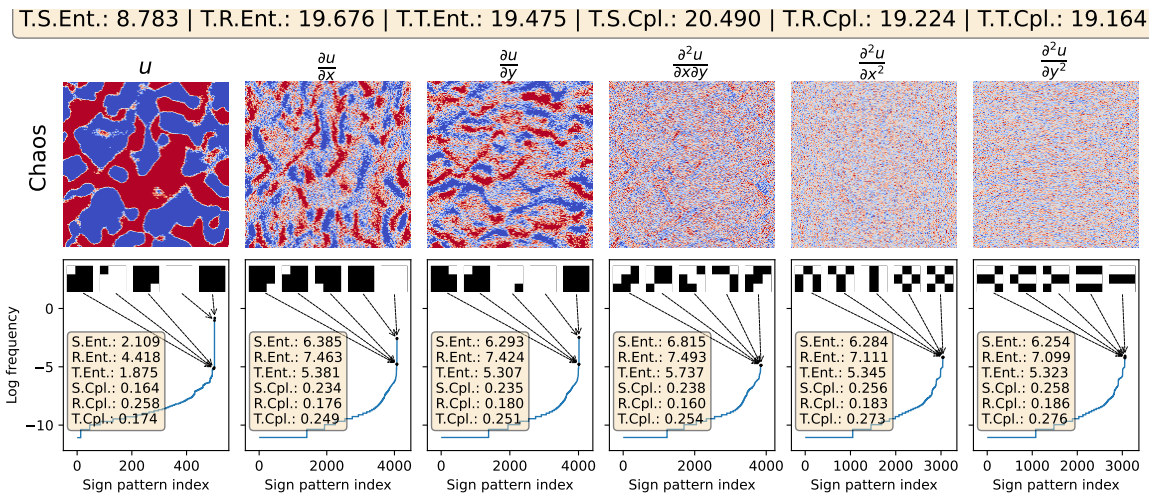


(a) Nano Holes

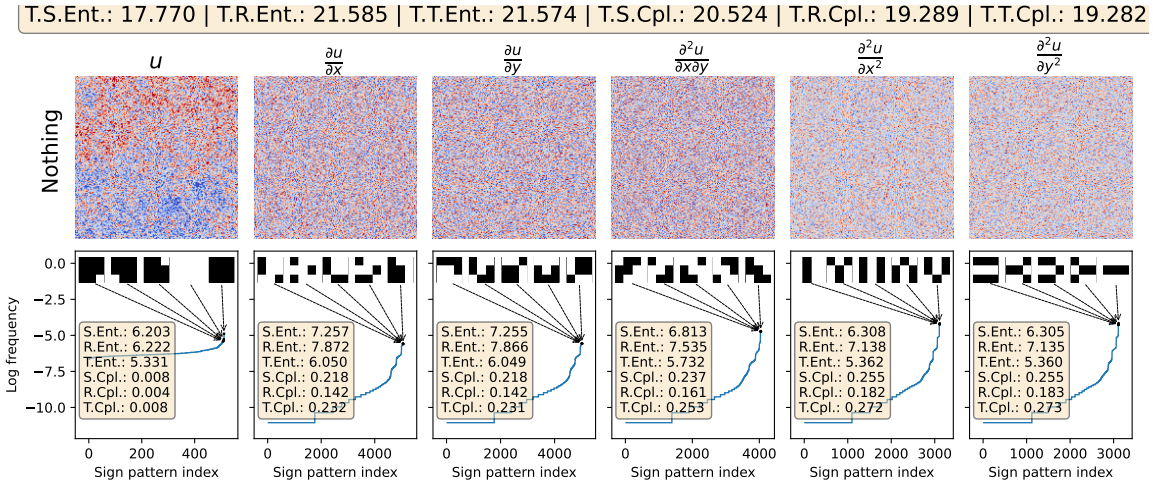


(b) Nano Stripes

Figure 2.15: Our original generalization of permutation entropy to the multidimensional setting – "Taylor entropy" –, taking into account local information, applied to selected samples of SEM fields. In each subfigure, the top row show the sign of the indicated field (the first, after subtracting the mean): red for > 0, white for 0, blue for < 0. The second row shows the 3x3 sign pattern distribution of the fields above in log scale. The small inset squares are the 5 most frequent patterns, with white < 0 and black > 0 (gray indicates 0, but it is not in the most frequent patterns), the arrows pointing at their location in the histogram. Inset boxes indicate Shannon, Rényi, and Tsallis entropies (in bits) and complexities (in bits^2), of the respective distributions. The suptitle shows the aggregate Taylor measures (cf.Sec. 2.6.6), in the same order. We set $q = 1.05$ and $\alpha = 0.5$ for Tsallis and Rényi measures, respectively.

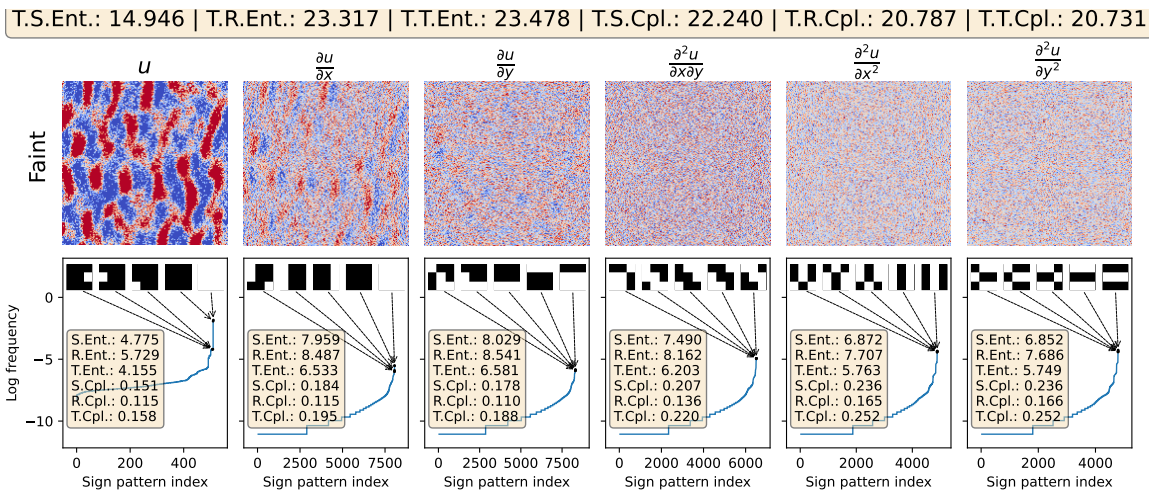


(a) Chaos

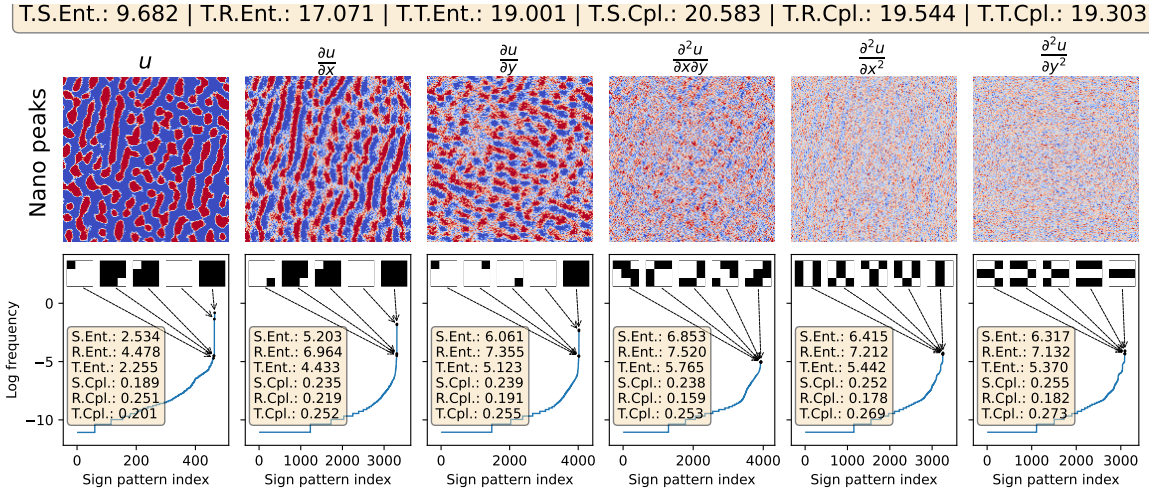


(b) Nothing

Figure 2.16: Our original generalization of permutation entropy to the multidimensional setting – "Taylor entropy" –, taking into account local information, applied to selected samples of SEM fields. In each subfigure, the top row show the sign of the indicated field (the first, after subtracting the mean): red for > 0, white for 0, blue for < 0. The second row shows the 3x3 sign pattern distribution of the fields above in log scale. The small inset squares are the 5 most frequent patterns, with white < 0 and black > 0 (gray indicates 0, but it is not in the most frequent patterns), the arrows pointing at their location in the histogram. Inset boxes indicate Shannon, Rényi, and Tsallis entropies (in bits) and complexities (in bits^2), of the respective distributions. The suptitle shows the aggregate Taylor measures (cf.Sec. 2.6.6), in the same order. We set $q = 1.05$ and $\alpha = 0.5$ for Tsallis and Rényi measures, respectively.

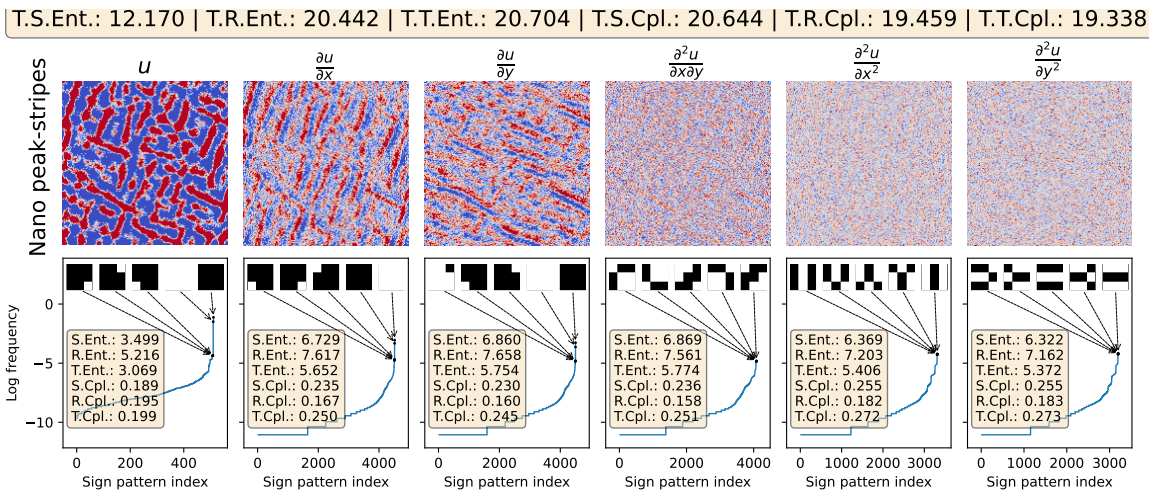


(a) Faint



(b) Nano Peaks

Figure 2.17: Our original generalization of permutation entropy to the multidimensional setting – "Taylor entropy" –, taking into account local information, applied to selected samples of SEM fields. In each subfigure, the top row show the sign of the indicated field (the first, after subtracting the mean): red for > 0, white for 0, blue for < 0. The second row shows the 3x3 sign pattern distribution of the fields above in log scale. The small inset squares are the 5 most frequent patterns, with white < 0 and black > 0 (gray indicates 0, but it is not in the most frequent patterns), the arrows pointing at their location in the histogram. Inset boxes indicate Shannon, Rényi, and Tsallis entropies (in bits) and complexities (in bits²), of the respective distributions. The suptitle shows the aggregate Taylor measures (cf.Sec. 2.6.6), in the same order. We set $q = 1.05$ and $\alpha = 0.5$ for Tsallis and Rényi measures, respectively.



(a) Nano Peak-Stripes

Figure 2.18: Our original generalization of permutation entropy to the multidimensional setting – "Taylor entropy" –, taking into account local information, applied to selected samples of SEM fields. In each subfigure, the top row show the sign of the indicated field (the first, after subtracting the mean): red for > 0, white for 0, blue for < 0. The second row shows the 3x3 sign pattern distribution of the fields above in log scale. The small inset squares are the 5 most frequent patterns, with white < 0 and black > 0 (gray indicates 0, but it is not in the most frequent patterns), the arrows pointing at their location in the histogram. Inset boxes indicate Shannon, Rényi, and Tsallis entropies (in bits) and complexities (in bits^2), of the respective distributions. The suptitle shows the aggregate Taylor measures (cf.Sec. 2.6.6), in the same order. We set $q = 1.05$ and $\alpha = 0.5$ for Tsallis and Rényi measures, respectively.

2.7 Experimental section

In this section, we show the application of the aforementioned measures of complexity to (i) a panel range of femtosecond laser induced patterns with constant N and varying laser fluence and time delay between pulses and (ii) five different increasing N series (cf. Figure 2.19).

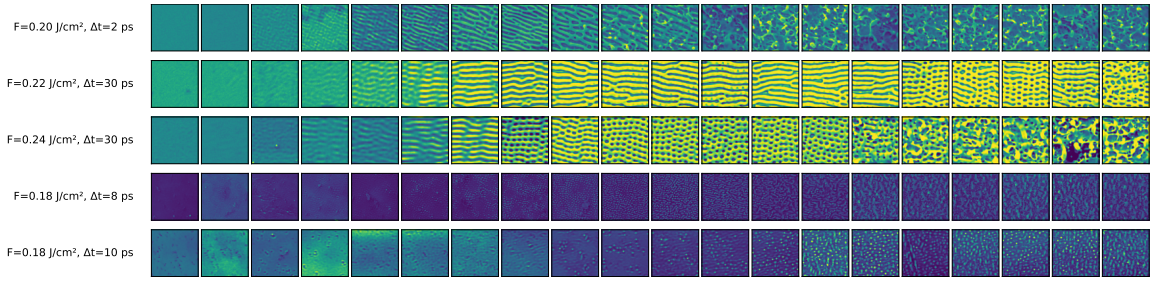


Figure 2.19: Visualization of SEM (Scanning Electron Microscopy) field samples for different laser parameters. Each row represents a series of samples generated with specific laser parameters, as indicated by the y-axis labels, at increasing values of N. Each column represents a sample taken at equal intervals within the series: in steps of 2, from 0 to 38 for all series except the second from the bottom, which is sampled from 10 to 48

As for (i), it provides a dictionary of sorts , which allows one to understand the behavior of the different measures of complexity where applied to SEM images. We present an exhaustive list in the hope that they will prove useful to the researcher trying to identify the measure of complexity that quantify disorder or complexity in a certain way.

With respect to (ii), the idea is to, as mentioned in Sec. 2.1, to investigate which of the measures of complexity that we examined, if any, allow us to take Bennet's stance, who functionally defines complexity as "whatever increases [or decreases] when something self-organizes". Conversely, the measure that increases in the case of self-organization of laser induced nanopatterns can also be used to characterize self-organization in this setting. The good news is that we identified a number of measures that monotonically increase or decrease when nanopatterns self-organize. What arguably stands out with respect to these measures is that (a) most show high variance when applied to real SEM patterns (b) the rate of increase/decrease differs for different series (which is not surprising) and changes from measure to measure.

This much being said, our newly proposed Shannon Taylor entropy arguably captures this researcher's intuitive notion of complexity the best (cf. Fig. 2.34), decreases during self-organization and shows comparatively low variance as shown in Fig. 2.36, and certainly merits further investigation. One can study, for example, the role of the parameters in Rényi and Tsallis entropies in the quality of the respective Taylor entropy, or the introduction of higher order derivatives. Another interesting question is how each of the derivative terms evolves with N. Finally, we would like to investigate the behavior of this measure in more controlled conditions, i.e. artificial data, in particular those generated by the Swift-Hohneberg equation which, we shall see in Chapter 3, is an apt model of pattern formation. For conciseness, we present a single example for each of the measures of complexity that we examined. The full list can be found in Appendix A.1.

2.7.1 Gray levels complexities

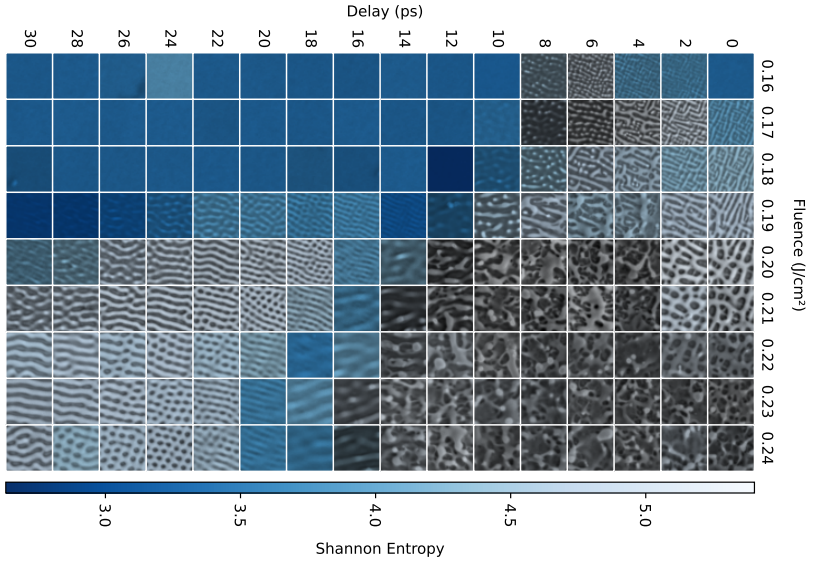


Figure 2.20: Shannon entropy of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

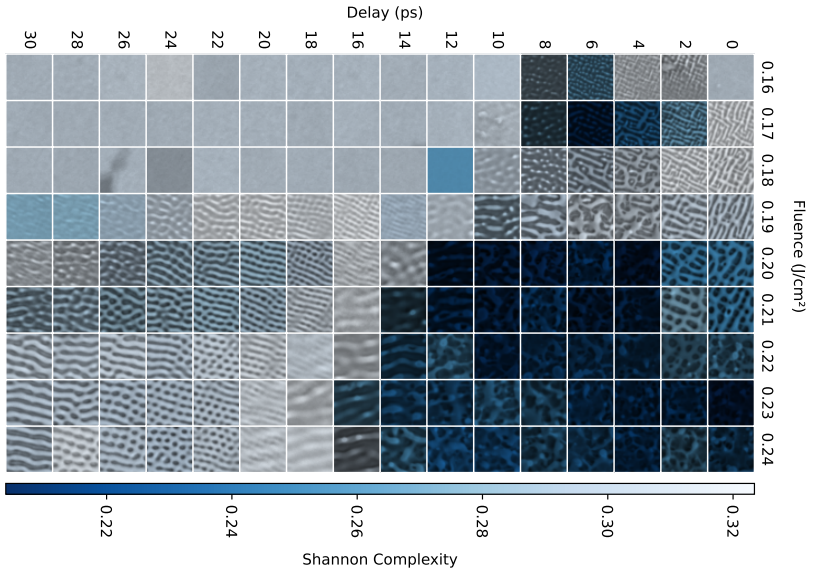


Figure 2.21: Shannon complexity of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

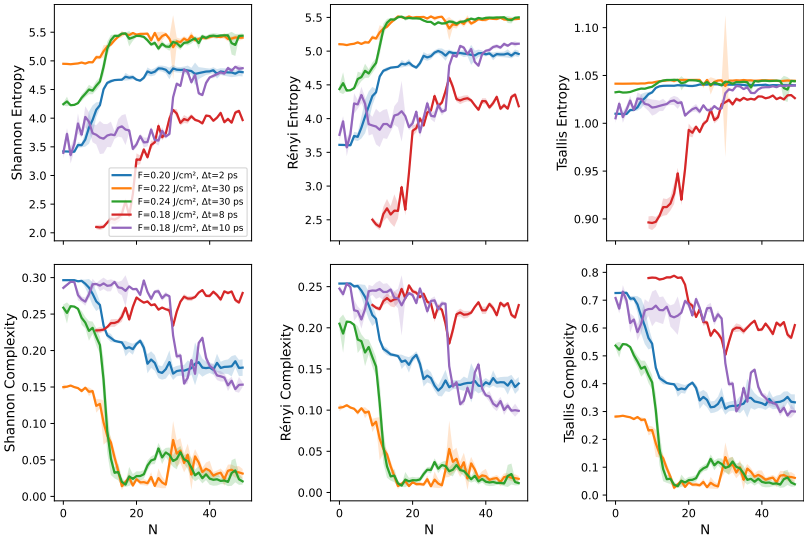


Figure 2.22: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) gray level entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image. All entropies increase up to a certain N, then stabilize: exception being the Purple series ($F = 0.18$ J/cm^2, $\Delta t = 10$ ps), with a great variety of different structures that form during the dynamics, ranging from holes to hexagons to chaos.

2.7.2 Gray level runs complexities

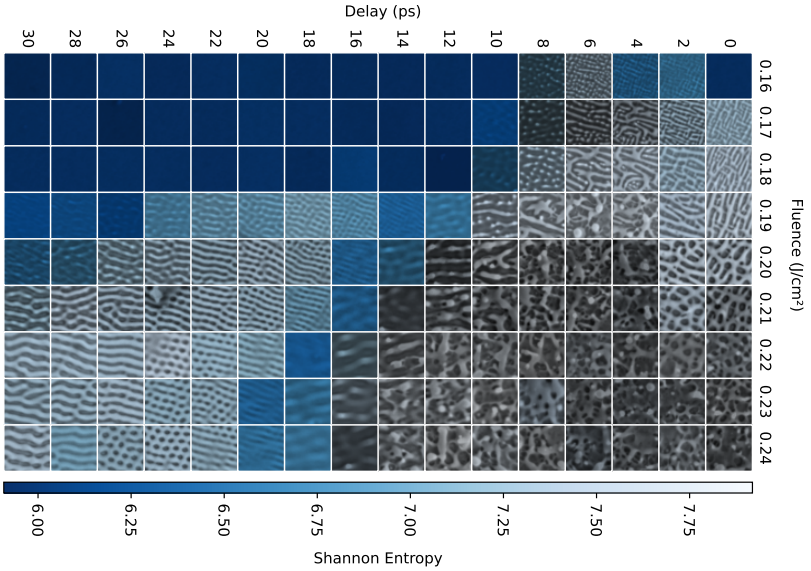


Figure 2.23: Shannon entropy of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

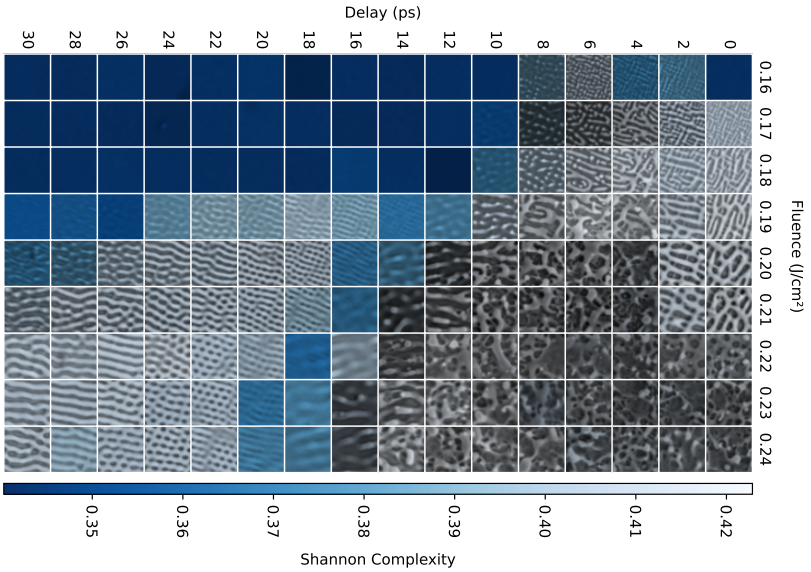


Figure 2.24: Shannon complexity of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

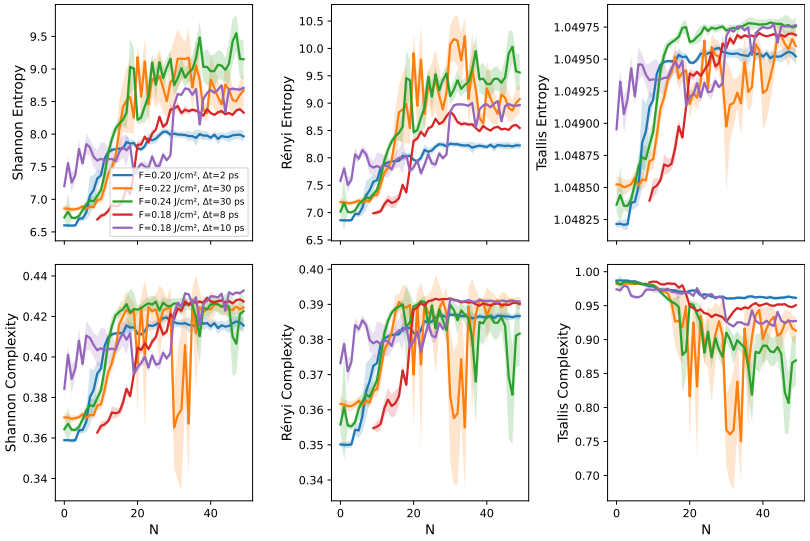


Figure 2.25: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) glr2 entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

2.7.3 Fourier spectrum complexities

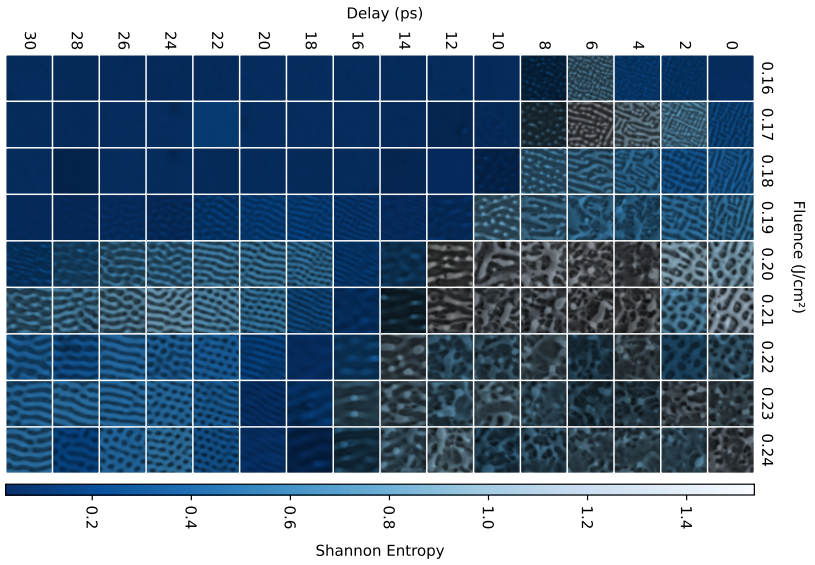


Figure 2.26: Shannon entropy of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

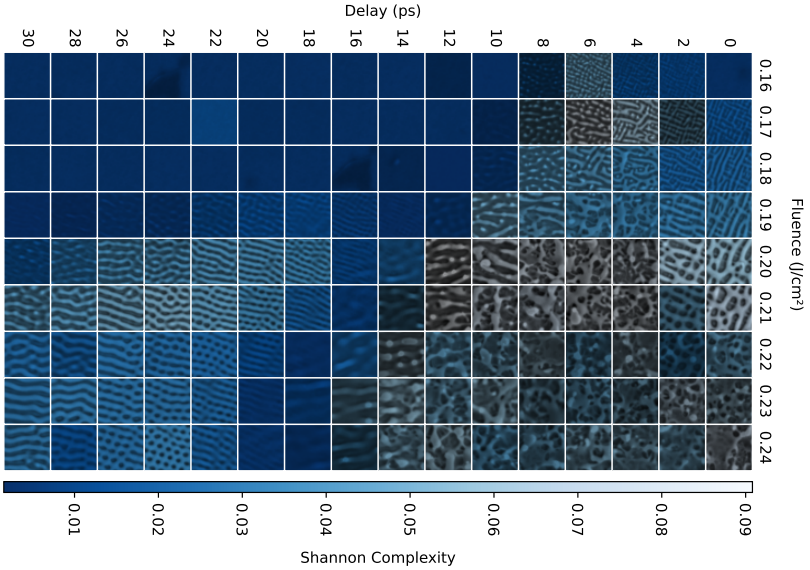


Figure 2.27: Shannon complexity of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

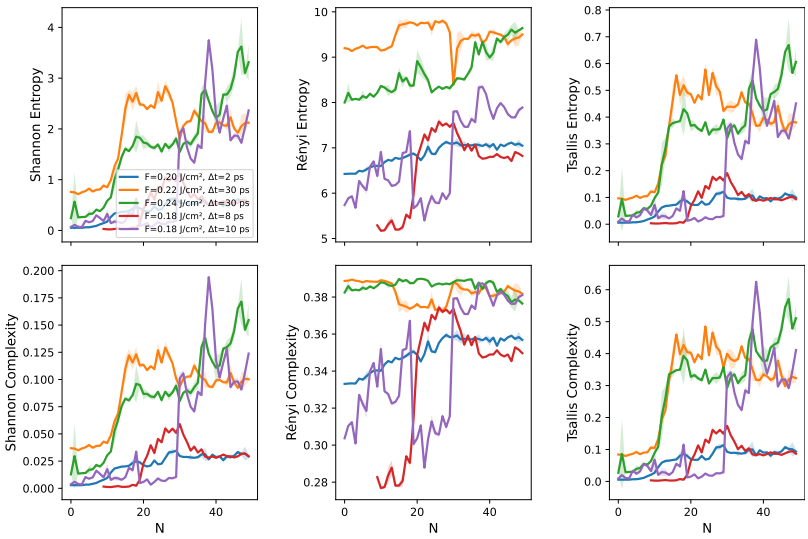


Figure 2.28: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) power spectral entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

2.7.4 Lempel-Ziv complexities

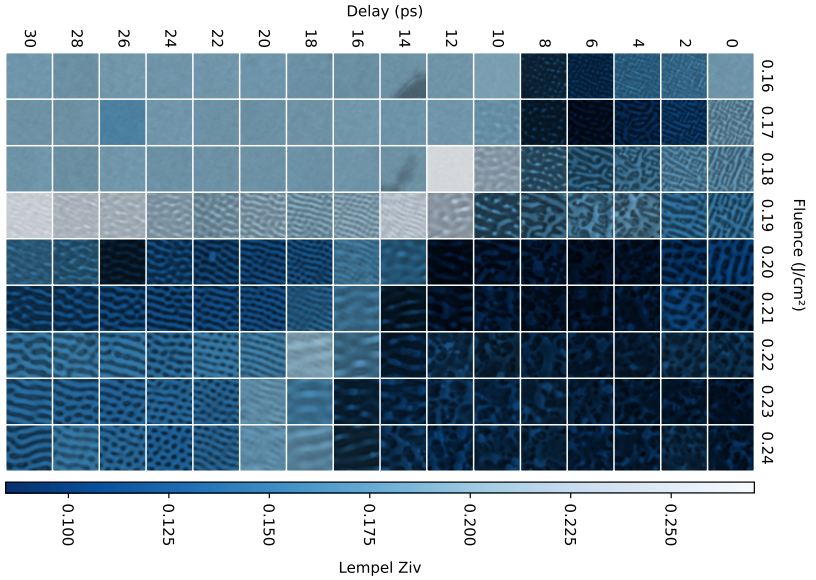


Figure 2.29: Lempel-Ziv complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

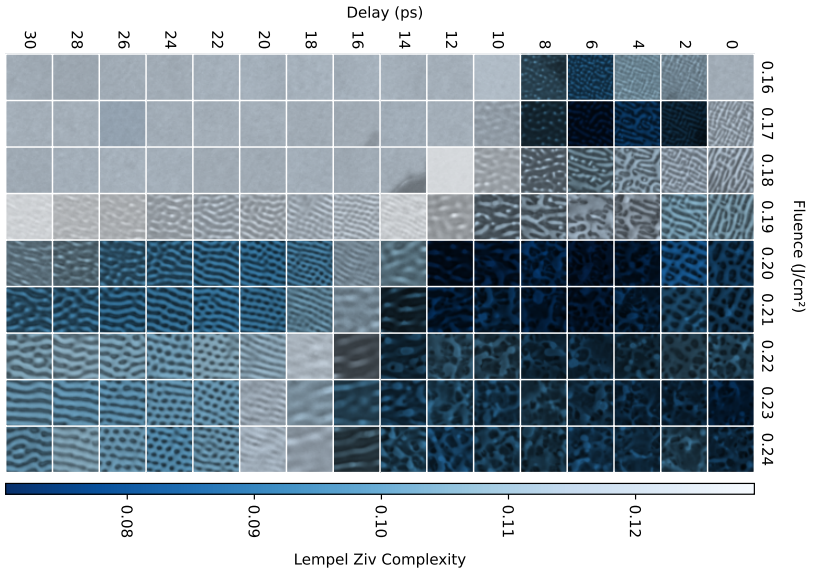


Figure 2.30: Intensive Lempel-Ziv complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

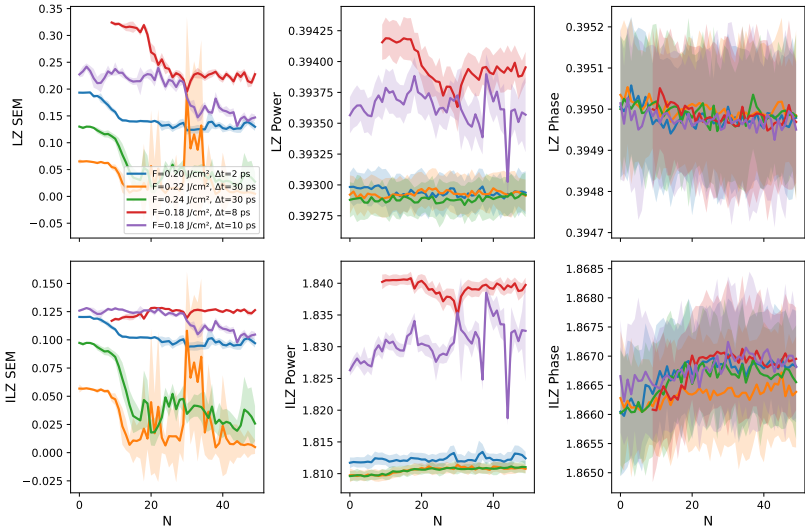


Figure 2.31: Comparison Lempel-Ziv complexities on the original SEM images, their Fourier power spectra, and phase spectra(top row) and intensive complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

2.7.5 Cross-patch similarity

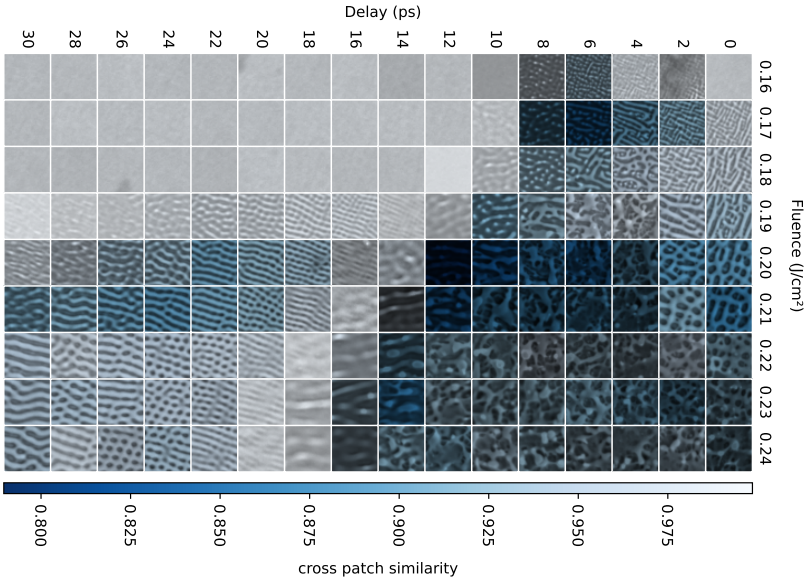


Figure 2.32: Cross-patch similarity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization. Each patch is a square with side length of 56 pixels at random orientations.

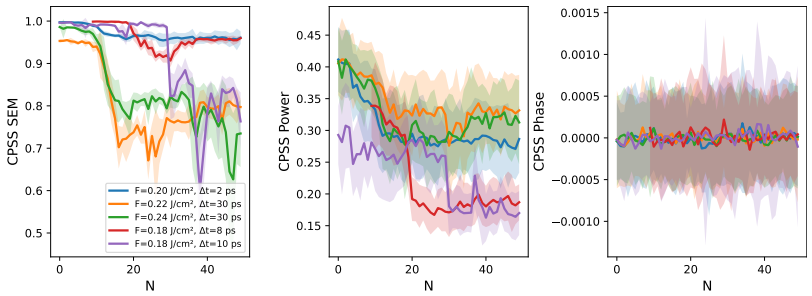


Figure 2.33: Comparison of Cross-patch similarity of the original SEM images, their Fourier power spectra, and phase spectra, for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image. Each patch is a square with side length of 56 pixels at random orientations.

2.7.6 Taylor complexities

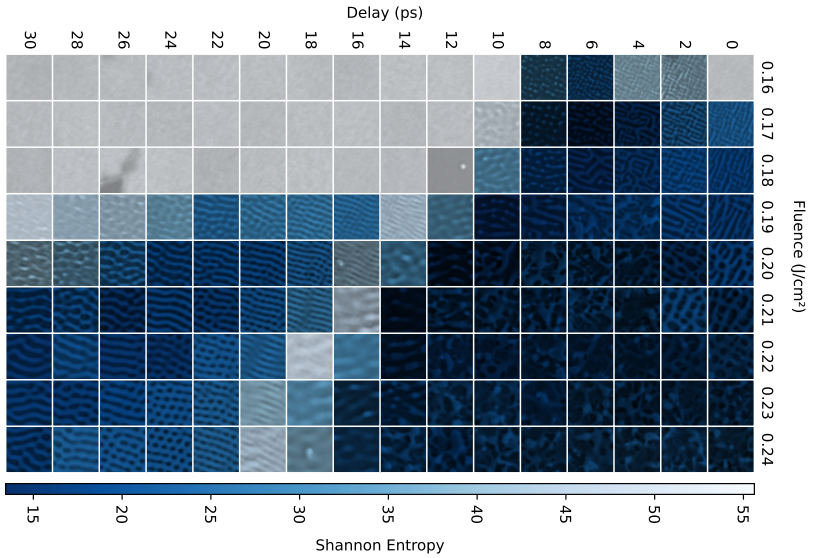


Figure 2.34: Taylor Shannon entropy of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

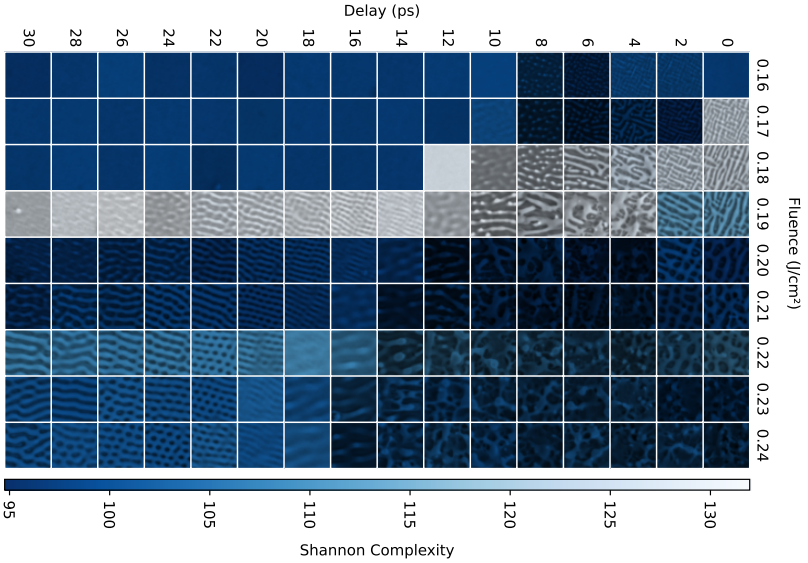


Figure 2.35: Taylor Shannon complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

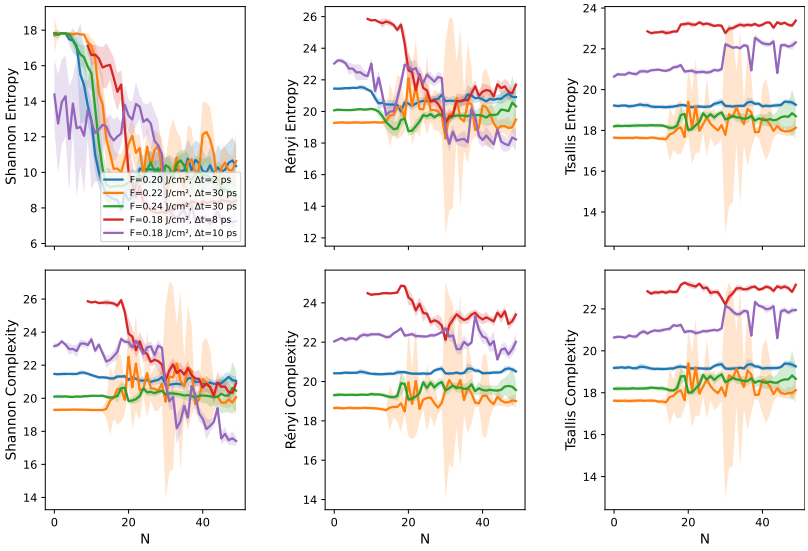


Figure 2.36: Comparison of Taylor-Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. 2.19 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

Chapter 3

Learning Complexity to Guide Light-Induced Self-Organized Nanopatterns

3.1 Introduction

Self-organization is prevalent in Nature. It is responsible for the interesting patterns and structures that we observe in systems outside equilibria, from the clouds of Jupiter to how the leopard got his spots. Without self-organization, we would observe mostly disordered states with no discernible structure [CH93]. Laser-irradiated surfaces are a chief example of a self-organizing system, as one observes coherent, aligned, chaotic and complex patterns that emerge at the microscale and the nanoscale [Rud+20]. These patterns are of great practical interest, with a number of potentially groundbreaking optical, hydrophobic and microbiological applications suggested in the literature [VG15; Ran+10; Bon+16; Grä20]. Laser texturing can be used as a method to reduce bacterial colonization of dental and orthopedic implants [Cun+16], as it can change surface wettability [VG15; Fad+11] and morphology, the interplay of which affect antibacterial properties [Lut+18] of the surface.

Laser induced periodic structures on stainless steel surfaces were also shown to act as a surface grating that diffracts light efficiently [Dus+10; Yao+12; Gua+17]. Since the orientation of the induced structures depends strongly on laser parameters, and it is possible to have multiple orientation structures with spacial overlap, surfaces can be decorated in such a way that different colors and patterns appear when white light is irradiated on the surface from different directions or even selectively displayed using structural color, with applications to encryption and anti-counterfeiting.

Importantly, laser texturing is reliably reproducible, meaning that these groundbreaking applications have the potential to be turned into industrial processes. In this respect, laser texturing has significant advantages over other surface functionalization methods such as electrochemical etching [Li+15], for example: laser induced structures can be generated in a simple, single-step process [Bon+16] that is reliable, reproducible and scalable, applicable to a great variety of materials [AGK14], and to large surfaces [Gni+17].

Put simply, under controlled experimental conditions, observed patterns are a function of the

laser parameters. This functional dependence is complex, with abrupt boundaries between pattern types for continuous variation of laser properties [Abo+20]. Pattern features of interest, such as rotational invariance, characteristic length, and feature height, for example, also show interesting dependence on laser parameters [Nak+21a; Abo+20; Rud+20] and novel structures that can be reproduced reliably have recently been observed [Nak+21a; Nak+22] (cf. Sec. 3.2 for a description of the experimental setup).

The variety of patterns and their potentially groundbreaking applications, combined with the controllability and reproducibility of laser-induced structure formation, motivate an exhaustive search of laser parameter space. This search is unfortunately impractical, as each experimental manipulation is costly and time-consuming. Having a model to guide the initialization of laser parameters would thus be of great interest.

A first solution to guide laser parameter selection would be to have a physical model explicitly relating pattern characteristics and laser parameters. This model, is however, not available. While a hydrodynamic process has recently been proposed to explain the hexagonal patterns observed in experiments [Abo+20], the full picture is complex and strict conditions are required for the process to occur upon ultrashort laser irradiation [Rud+20] (cf. Sec. 3.2.3 for an overview of this model).

The photoexcited material evolves in a non-deterministic way due to stochastic surface roughness that may trigger local nonlinear optical response and collective thermomechanical response. In that far from equilibrium conditions, a deterministic approach is able to explain specific nanostructuring regimes [Nak+22] but fails to predict the coexistence and the transition between several nanopatterns (cf. Sec. 3.2.4) for our original approach that offers an explanation for the coexistence between nanopatterns).

In the absence of an explicit physical model, a second solution to guide laser parameter selection is to use machine learning, which can produce data-driven models with remarkable results. The quality of these results, however, depends on the quantity of training data available. More precisely, drawing from the language of statistical learning theory [FHT+01], let h be some hypothesis (a model) chosen from some class of hypotheses $\mathcal{H}$ of a given complexity C[1], which is chosen by using some algorithm based on minimizing the expected loss (intuitively, expected error) on training data X_{train} drawn from an unknown distribution D. Then ΔE, the expected difference in error that we make by evaluating our model on unseen test data X_{test} drawn from the same distribution is bounded above by g, a function of the model complexity and the number of training examples $|X_{train}|$. Crucially, it can be shown that g is an increasing function of model complexity C and $g \to 0$ as $|X_{train}| \to \infty$

$$\Delta E \leq g(C, |X_{train}|), \tag{3.1.1}$$

which means that to keep ΔE small with little data, one should keep the model complexity small. In this sense, the complexity of the model is bounded by the quantity of data: with little data, we can only effectively learn simple models. This is precisely the case with laser induced pattern experimental data, since the difficulty in data acquisition both motivates and hinders the construction of a machine learning model. This situation is far from being exceptional, as most "natural" scenarios are neither in the high data regime [Sil+16], nor in the high model, little data regime [Har+13].

There are a variety of methods and techniques to learn from data in this case, namely by integrating physical information to guide the ML model. This collection of techniques is generally grouped in under the "Physics-guided" or "Physics-informed" machine learning topic where, among

[1] Such as Vapnik-Chervonenkis dimension, Rademacher complexity, uniform stability or algorithmic robustness to cite a few.

other things, some approaches aim to integrate physical knowledge in the form of a PDE to solve a certain task [Kar+21; Wil+20; Jia+21; Um+20; YPK21]. Our work falls into the scope of this scenario. As a physical model of the convective process that is at the origin of the observed physical structures, we use in this work the *Swift-Hohenberg* partial differential equation (PDE) on the plane [SH77], a simple and well-studied model of complex pattern formation under Rayleigh-Bénard convection [CH93]. We leverage this PDE because in spite of it being a considerable simplification with respect to the actual process taking place in laser irradiated surfaces, it is still compatible with the physical situation, as originally shown in Sec. 3.2.4. Moreover, the Swift-Hohenberg equation can be seen as maximally symmetric model of pattern formation, as shown is Sec. 3.5.3, and most of the symmetries involved in the derivation are indeed present in the initial stages of laser-matter interaction before thermalization.

Pattern-like solutions of Swift-Hohenberg (SH) equation are remarkably similar to the ones that we can observe in the irradiated surfaces Scanning Electron Microscope (SEM) images, as can be seen in Figure 3.1. In spite of its simplicity and longevity, the variety of pattern-like solutions of SH equation still makes it a topic of active research [EC20].

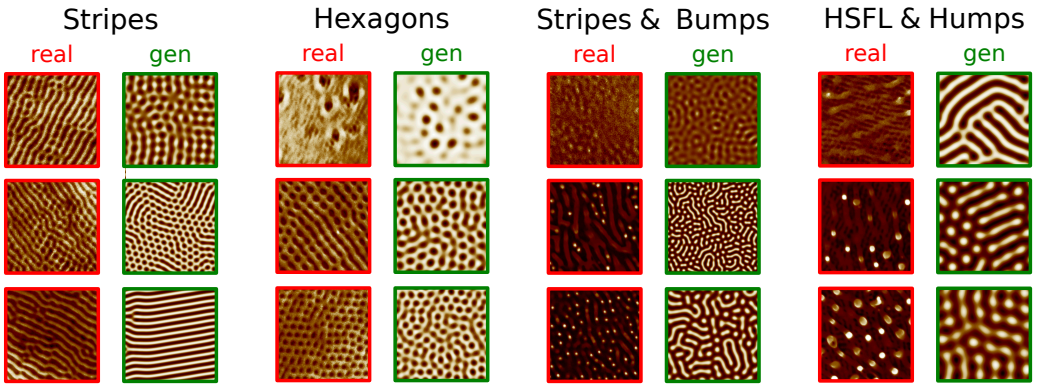


Figure 3.1: Comparison between real SEM images (red) and SH-generated images (green). SH-generated images are able to reproduce a variety of patterns (e.g. stripes, hexagons, bumps, hsfl, humps) and scales. A couple of model simplifications can be observed in this comparison: first, since the SH model is an isotropic model, global symmetries are only apparent (e.g. oriented stripes), whereas SEM images retain some measure of global symmetry from laser polarization. Second, since the SH model is a single scale model (in the sense that there is a single critical wavelength), it can only match a single pattern, even for SEM images that contain a superposition of patterns (such as e.g. 7th column, top image).

The main objective of this chapter is to design new physics-guided ML techniques that allow us to integrate **partial** physical information and learn with **few data** the relationship between patterns and laser parameters *given that the patterns were produced via a SH convective instability.* While the complexity of the resultant model is still bounded by the same quantity of data, we expect it to be more faithful. To be precise, although the difference in error ΔE in (3.1.1) might be small, the minimum error that can be attained in $\mathcal{H}$, which measures the quality of our model, might still be quite large. But if integrating physical knowledge $\mathcal{H}$ makes it more compatible with

the physical situation (in this sense more faithful), the minimum error will be smaller[2].

In situations where similar patterns are observed, there is considerable information contained in the *parameters* of the Swift Hohenberg equation, which determine the *type* and general features of the pattern solutions for a range of initial conditions. In this sense, for the purpose of predicting novel patterns, there is too much physical information in the SH solutions, and a feature transformation F exists such that in the image of F, patterns can be effectively described using model parameters alone (see Fig. 3.2 for details).

As we shall see in the sequel, this key insight informs our contribution which is five-fold: **(i)** we present a Bayesian inference formulation for solving the dual inverse problem of estimating state and model parameters in the case of self-organization, **(ii)** we design an efficient solver of SH PDE allowing the generation of a large amount of SH (parameters-patterns) pairs of data **(iii)** from this large dataset, we learn a differentiable-Neural Network (NN) surrogate of the SH PDE, (**iv**) we leverage this pre-trained NN, on the one hand, and the SH (parameters-patterns) pairs, on the other, to learn two alternative end-to-end models from the laser parameters to the patterns obtained by laser irradiation; **(v)** we conduct experiments showing a good agreement between the generated images and experimental data for some parameter ranges, which can serve as a guide for laser initialization and suggest improvements to the SH model.

The rest of this chapter is organized as follows: we begin, in Section 3.2 by briefly presenting the experimental setup involved in the creation of nanopatterns on Nickel. We then proceed by schematically describing the two-temperature physical hydrodynamic model of laser-matter interaction presented in [Rud+19a; Rud+20], which is the state of the art. We then originally show (unpublished material) that the ionic fluid follows Boussinesq-like equations after the faster electronic dynamics has been allowed to equalize.

In Section 4.2 we frame the problem of learning novel laser patterns by integrating partial physical information in the form of a PDE and discuss related work. In Section 3.4 we present a framework for solving the dual inverse problem with few data by integrating partial physical information in the context of systems with self-organization. In Section 3.5 we apply this framework to the specific case of learning novel laser-induced patterns on monocrystalline Nickel (Ni) at the nanoscale: we begin by presenting the SH equation as partial model of the physical situation. To do so, we start by summarizing the original derivation of the equation in Section 3.5.2, which was done in the case of Rayleigh-Bénard convection in the Boussinesq approximation. Our original result above regarding the dynamics of the ionic fluid then implies that multi-double pulse dynamics of the laser-matter interaction with crossed polarization follows the Swift Hohenberg equation. As the Swift-Hohenberg equation is a model for pattern formation, we have thus been able to explain the mechanism of nanopattern formation by femtosecond laser irradiation. We also present an alternative derivation of the Swift-Hohenberg equation as a maximally symmetric model of pattern formation, which further justifies its use as a simplified model. We show that the Swift Hohenberg equation has potential dynamics, and for completeness, describe how Swift-Hohenberg-like equations can be derived from a potential. In order to use SH as a simplified model of the physical situation for learning novel patterns, we need to to be able to solve it efficiently. We thus proceed by presenting a pseudo-spectral second-order solver of the SH equation combining accuracy and speed in Section 3.5.5; we continue by discussing, in Section 3.5.6, a particular feature mapping that we chose for our problem, which allows considerable simplification, and that we validate via a quality measure based on expert clustering results in Section 3.4.4. And finally, we present two

[2] As an extreme example, consider learning the sine function, which has an infinite power series expansion (complex) and a one-term expansion in Fourier series (simple).

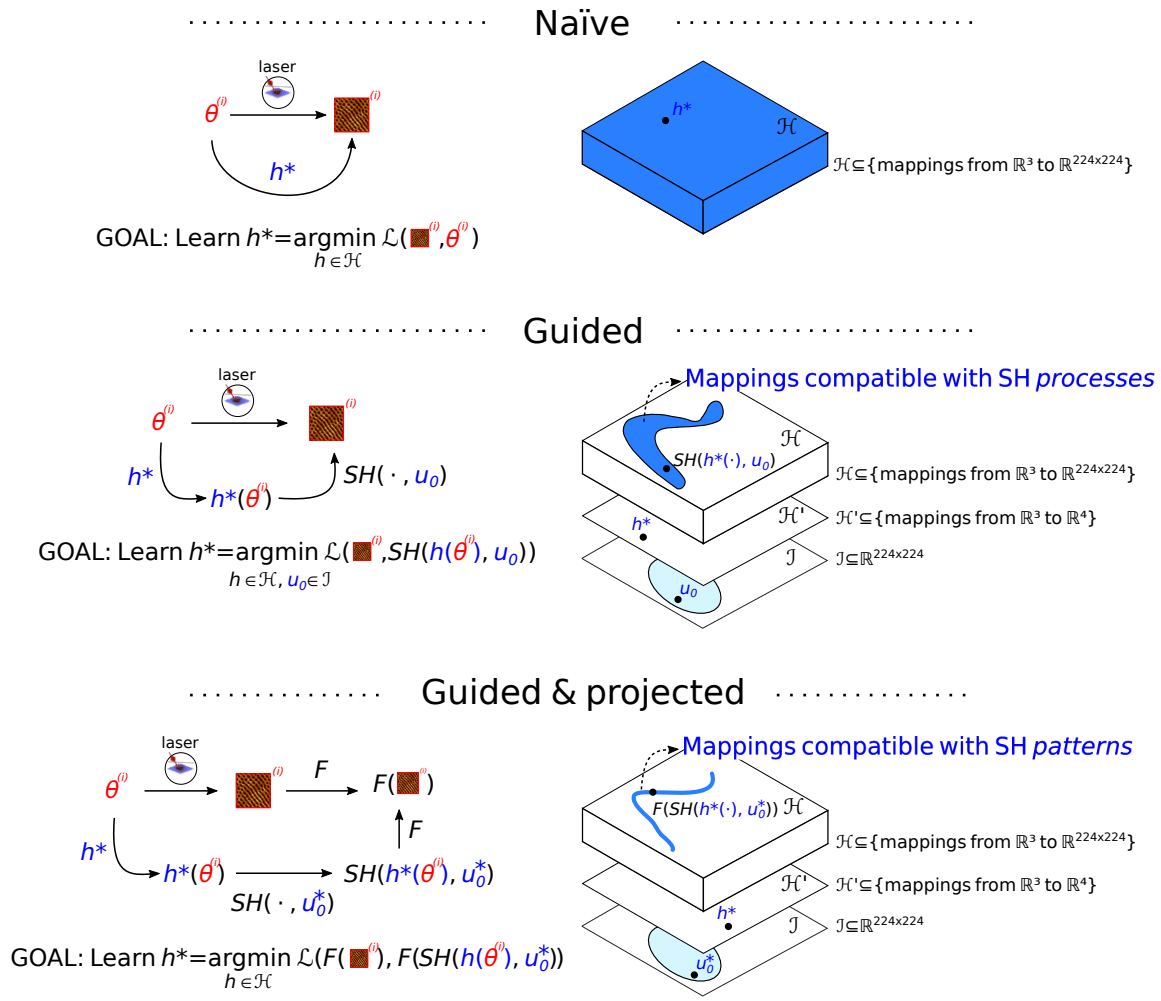


Figure 3.2: **(top)** The naive approach to learning the relationship between laser parameters $\theta \in \mathbb{R}^3$ and fields $\in \mathbb{R}^{224\times224}$ (224 by 224 pixels SEM images) is to seek the minimizer h^* of the loss $\mathcal{L}$ in the large set $\mathcal{H}$. With few observations, this is unfeasible. **(middle)** Physical knowledge can guide the search by restricting it to mappings that are compatible with the SH model — the much smaller subset of $\mathcal{H}$ in blue. The learned mapping is now only between $\theta \in \mathbb{R}^3$ and SH parameters $\in \mathbb{R}^4$. We still have to minimize over initial conditions $u_0 \in \mathcal{I}$, to which we do not have experimental access. Again this is unfeasible. **(bottom)** Since we are only interested in general pattern features, and the SH model is based on a self-organization process, a feature mapping F exists such that in the image of F, patterns can be described using model parameters alone. We safely ignore initial conditions (e.g. choosing u_0^*) and seek the minimizer over mappings that are compatible with SH *patterns* — the much smaller subset of $\mathcal{H}$ in blue. This is feasible with few data.

alternative models to learn the relationship between laser parameters and SH parameters, which will allow us to predict novel patterns in Section 3.5.7. In the next Section 4.5 we present and

discuss experimental results; notably, we see that pattern features are correlated and that more than one SH process may be at play, which is a new physical insight. We conclude in Section 4.6 and discuss future research.

3.2 Self-organized nanopatterns formation

The emergence of instabilities and symmetry breaking leading to the formation of coherent structures, is one of the most fascinating aspects of the complex dynamics governing light-surface interaction [Her+98; Shi+03; Ild+17]. When a randomly rough surface is subjected to ultrafast laser pulses, it enters a far-from-equilibrium state due to the repeated absorption of pulsed optical fields. As a result, the surface exhibits spontaneous spatial organization, which is oriented by energy gradients generated by laser polarization, giving rise to laser-induced periodic surface structures (LIPSS) [Sip+83]. These structures form under far-from-equilibrium conditions and can be triggered by capillary waves, convection rolls, and thermoconvective instabilities, [Kei83; YSV84; Tsi+16; Rud+20] which persist through dissipative structures [PV63]. Eliminating the prevailing laser polarization effects reveals puzzling patterns emerging from a sequence of instabilities, inducing different types of complex patterns, ranging from chaos to six-fold symmetries [Nak+21a]. The photoexcited matter undergoes a transition from a disordered state to a more coherent one, referred to as a *strange attractor* in the phase space of nonlinear dynamics. This transition results in a metastable state, defining a self-organization structuring regime. Through this self-organization process, the material surface can be sculpted seamlessly, enabling nanoscale manufacturing [Nak+22]. Understanding the selection mechanisms involved in this morphogenesis to gain control over the uniformity, symmetry, and size of the resulting surface patterns is a major research theme in laser processing for photonics metasurfaces, biomimetics, or catalysis functionalization. [Str+20; OMY20]. To apply statistical inference approaches to complex systems and achieve generalizability, advanced physics-guided machine learning strategies are essential.

Upon laser irradiation, a hazy boundary separates self-organized and organized surface patterns. When a material is exposed to sufficiently intense laser irradiation, it tends to organize along the stationary electromagnetic fields due to scattered/excited waves [Sip+83; Rud+19b] and self-organize in response to the random fluctuations of light absorption with a symmetry breaking with respect to polarization [Var+06; Abo+19]. Light-oriented and self-assembled dynamical processes are inherently superimposed, and surface topographies evolve spatio-temporally towards equilibrium patterns that result from a complex competition between free energy dissipation imposing entropy production and spontaneous ordering.

Consequently, any preexisting or transient organization can be disrupted by random perturbations, which can be amplified by positive feedback to lead the system towards new patterns.

Ultrafast laser texturing has recently been used to obtain deep sub-wavelength periodic patterns, which raises questions about the relevant electromagnetic processes that drive the formation of these patterns well below the diffraction limit [SC20; BG20].

Various types of 2D surface patterning have been reported, including patterns with oriented, triangular, hexagonal, labyrinthine, or chaotic symmetries [Qia+18; Fra+19; Abo+20; Mas+21], featuring both positive and negative reliefs such as humps, bumps, peaks, and spikes [Nak+21a]. To explain the remarkably uniform establishment of these patterns on the microscale independently from the oriented near-field optical effects on the random local nanotopography, a more global and collective perspective is required [Abo+20; Nak+21a]. Nanoscale fluid flows were shown to be

driven by a complex interplay between electromagnetic, internal and surface pressure forces which can become trapped due to the resolidification process [Abo+20; Rud+20].

The deterministic approach to predict the underlying optical coupling processes is limited because it requires the artificial integration of fluctuating conditions induced by surface roughness. Transiently formed structures can become unstable under nonlinear amplification and bifurcate into more complex patterns that are not accurately described by classical approaches like Navier-Stokes combined with Maxwell equations. Nonetheless, the complex pattern landscape has been experimentally explored and can be now compared with mathematical models dedicated to nonlinear system dynamics.

The Kuramoto-Sivashinsky approach has become a paradigm for describing pattern formation and spatiotemporal chaos on surfaces eroded by ion bombardment, which ultimately reproduces ripple formation and other organized patterns [BS10]. A similar approach was initially proposed for laser-induced nanopatterns, although a clear physical picture has yet to be established [Rei+12]. Along similar lines, the Swift-Hohenberg (SH) dynamics has been identified as a relevant candidate for representing the observed complexity of convective instabilities with spatiotemporal features, such as chaos, rolls, and hexagons [EVG92; CH93]. The SH approach has proven to be useful in identifying generic spatiotemporal dynamics of patterns in convective fluids [DPW94; ER00], as well as curvature- and stress-induced pattern-formation transition [Sto+15]. The SH approach was formally deduced from the Navier-Stokes equations in the Boussinesq approximation, with thermal fluctuation effects in a fluid near the Rayleigh-Bénard instability [SH77].

3.2.1 Experimental setup

We briefly describe the experimental setup in [Nak+21a] for completeness.

The experiment involves directing an ultrafast Ti:Sapphire laser onto a 10mm cube of monocrystalline Nickel (Ni) with a (100) orientation, which ensures uniform laser-induced structuring. To ensure a smooth initial surface, the sample undergoes both mechanical and electrochemical polishing to achieve a surface roughness below 5 nm, as confirmed by Atomic Force Microscopy (AFM). The laser setup includes a modified Mach-Zehnder interferometer, which is an optical device that splits the laser beam into two separate beams. These beams may travel along optical paths of different lengths, allowing for the introduction of a temporal delay between them. Importantly, the interferometer is adjusted to produce beams with perpendicular polarizations. The crossed polarization facilitates isotropic energy deposition on the sample, which is crucial for the formation of the specific patterns of interest.

After irradiation, the surface topography changes from the initial sub 5 nm uniform roughness. The resulting topography ranges from chaotic structures to very regular structures of different shapes and scale, as was observed using Scanning Electron Microscopy (SEM) (cf. Fig. 3.3.

Surface irradiation in double pulses was conducted a number of times N (*number of pulses*, with each pair of cross-polarized double pulses separated in time by a certain *time delay* Δt and with a combined total fluence F. As these parameters are allowed to change, so do pattern features, in an intriguing and highly non-linear way. As stated above, the dynamics of this change is not currently fully understood, as laser-matter interaction induces a highly complex dynamics over extremely short time-frames – too short to even allow image acquisition with standard experimental setups.

Tailoring nanotopographic features on a surface is a challenging task that has been successfully accomplished using ultrafast laser processes with time-controlled polarization strategies. Numerous regimes of LIPSS have been reported with various periodicities, heights, orientations, and sym-

metries depending on different polarization directions between the first $\vec{E_1}$ and second pulse $\vec{E_2}$, characterized by $\alpha = (\vec{E_1} \cdot \vec{E_2})$ in Fig. 3.3(a) [Bon+12; Wan+20; Abo+20; Nak+21a].

Figs. 3.3(b-d) present surface topographies measured by high resolution atomic force microscopy (AFM). A circular region with a diameter of 1 μm corresponding to the laser impact center was mapped in 3D (tilted) mode in Fig. 3.3(b-c) and in 2D for Fig. 3.3(d). To observe the significant role of temporal pulse splitting Δt in nanopatterns control, laser peak fluence F and N were kept fixed at 0.18 J/cm^2 and 25 respectively, as shown in Fig. 3.3(b). At $\Delta t = 8$ ps, organized nanopeak structures were observed with a high aspect ratio, a height of $\sim$ 100 nm and a diameter of $\sim$ 20 nm [Nak+22]. An extension of 2 ps in Δt modifies the observed patterns that turn into a different organization, a regime referred to as nanobumps [Nak+21a]. For $\Delta t = 15$ ps, a regime of nanohump generation is reached with a lower aspect ratio as the structures display a height of $\approx$ 10 nm and a diameter of $\approx$ 30 nm.

The role of laser fluence is revealed by fixing $\Delta t = 25$ ps and $N = 25$, as depicted in Figure 3.5.2(c). At $F = 0.18$ J/cm^2, a low-contrast nanopeak regime is formed, evolving into a nanostripe pattern with a slight increase in laser fluence increase to 0.20 J/cm^2. At $F = 0.22$ J/cm^2, a transition region is established, combining both stripes and cavities. Finally, at $F = 0.24$ J/cm^2, the surface is uniformly organized with hexagonally arranged nanocavities having a depth of $\approx$ 25 nm and a diameter of $\approx$ 30 nm. Both nanohumps and nanovoids result from hydrothermal flows guided by surface tension and rarefaction forces, leading to thermoconvective instability at the nanoscale, similarly to well-known Rayleigh-Bénard-Marangoni instabilities [Abo+20; Nak+21a; Vit+20; BPA00; TB95; Pea58; Mor+18b; SD83; Smi86; TFS15; Bus14; BT99; BV98; SP15; Rud+20]. Laser dose also plays a role, as positive feedback regulates pulse-to-pulse topographical transformations. As shown in Fig.3.5.2(d), at a fixed $F = 0.24$ J/cm^2 and $\Delta t = 8$ ps with different N, corresponding to the parameters of nanopeaks formation presented in Fig.3.5.2(a), three different surface organizations were observed. Pulse-to-pulse growth dynamics exhibits the transitions from convection cells ($N = 15$), to the creation of crests on the convection cells ($N = 20$). The nanopeaks grow on the edges of the crests to reach their optimal shape, concentration and organization at $N = 20$.

Before we set to combine the existing partial physical understanding of this dynamics with the available data, in the next section we shall present a brief overview of the state of the art model used in [Rud+20]. We shall then spend some time presenting an original analysis of the dynamics in the initial picoseconds (tens of picoseconds for Ni). We show that during these early moments, the dynamics can be described using a particularly simple model, the Swift-Hohenberg model, which has pattern-like solutions. This is the model that we shall ultimately use as "physical knowledge".

3.2.2 Rayleigh-Bénard convection

Rayleigh-Bénard convection [Ray16; Bén00] is a type of natural convection, and a cornerstone phenomenon in the study of pattern formation due to the high level of agreement between theory and experiment.

The setup for observing Rayleigh-Bénard convection involves a thin fluid layer, typically water or oil, with a thickness d, between two flat, horizontal plates in a gravitational field, assumed to be ideal heat conductors. The upper plate is maintained at a temperature T_0 that is lower than the upper plate's temperature $T_0 + \Delta T$. The temperature difference $\Delta T > 0$ drives the system out of equilibrium.

In the absence of a significant ΔT, the fluid remains static. However, as ΔT increases, buoyancy forces begin to act on the fluid, causing the warmer, less dense fluid at the bottom to rise. This

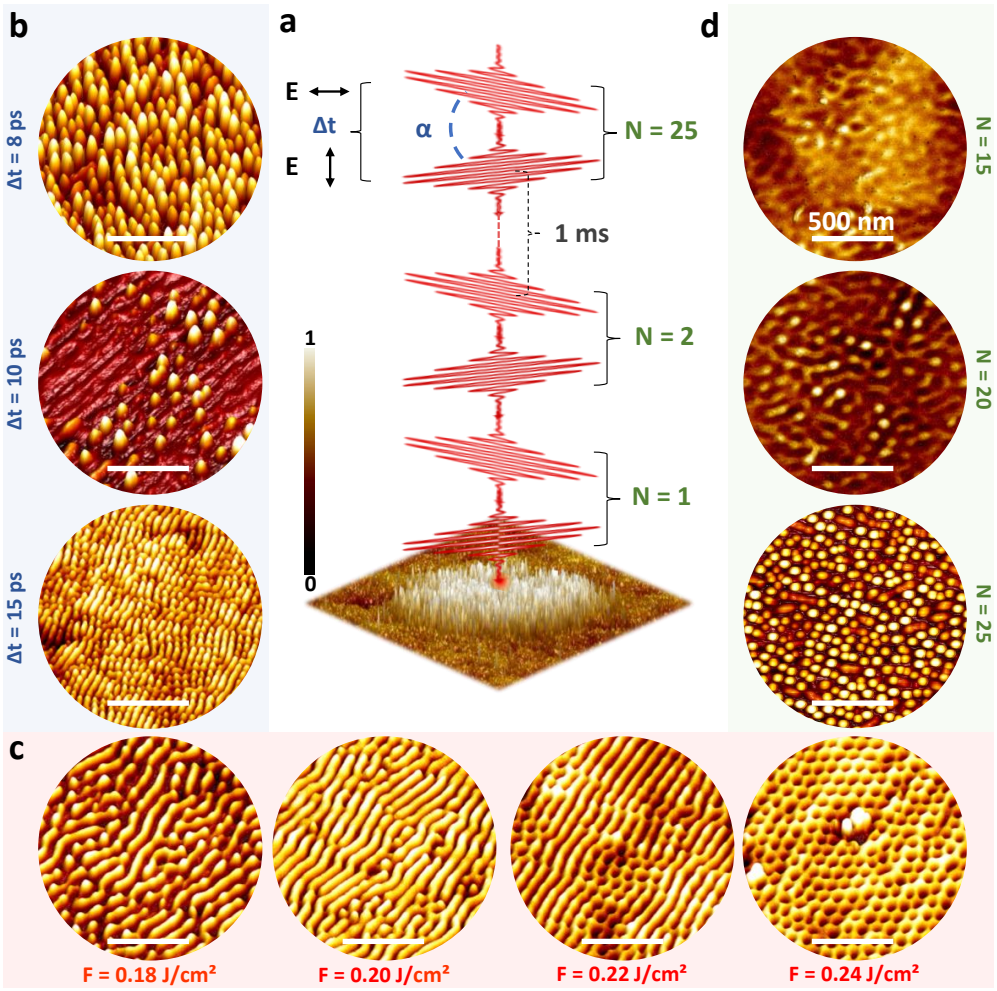


Figure 3.3: Adapted from [Nak+21a](a) Schematic illustration of experimental self-organization regimes induced by bursts of ultrafast laser double pulses. (b) Self-organized patterns of topography that develop varying time delays for a given F and N (AFM-3D mode). (c) Nanopatterns variation with respect to laser fluence at fixed Δt and N (AFM-3D mode). (d) Nanostructure growth by feedback at different number of pulses (AFM-2D mode), for a fixed Δt and F. The scale bars represent a length of 500 nm.

buoyancy force is given by $a\rho g\Delta T$, where a is the thermal expansion coefficient, ρ is the average mass density, and g is the acceleration due to gravity. These buoyancy forces are counteracted by dissipative forces due to thermal conduction and viscosity, represented as $vK\rho/d^3$, where v is the kinematic viscosity and K is the thermal diffusivity.

Convection occurs when the buoyancy forces exceed dissipative forces, a transition that is char-

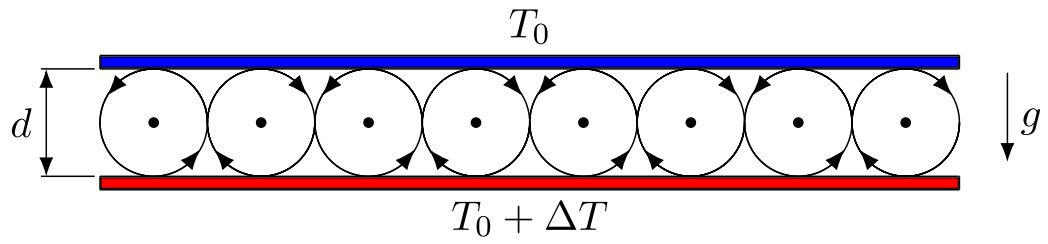


Figure 3.4: Schematic representation of Rayleigh-Bénard convection. The lower plate, shown in red, is maintained at a higher temperature $T_0 + \Delta T$ while the upper plate at a distance d above it, shown in blue, is at a lower temperature T_0. Convective rolls form in the fluid layer between the plates due to the temperature gradient.

acterized by a non-dimensional parameter called the Rayleigh number R, defined as:

$$R = \frac{ag\Delta T d^3}{vK} > R_c \approx 1708 \tag{3.2.1}$$

If all fluid were to rise, it would violate mass conservation. Hence, as some fluid rises, other descends, leading to the formation of convective patterns, the simplest form being convective rolls, schematically represented in Fig. 3.4. The size of these rolls is proportional to the fluid layer thickness, and they occur at a critical wavelength, characterized by a wave number $q_0 \approx 3.17/d$.

3.2.3 Hydrodynamics of nanopattern formation by laser irradiation

The complex interaction between matter and ultrashort laser is commonly modeled by using the well-known two-temperature model [AKP+74]. In metals, the electrons in the conduction band absorb laser energy and subsequently transmit it in the interaction volume due to their thermal conductivity. At the same time, the lattice is heated through electron-phonon collisions. The general intuition is that there are two essentially different dynamics: that of the fast, mobile electrons and that of the relatively slow phonons.

The fast electrons in the conduction band absorb laser energy, a process that can be modelled using the Maxwell equations for electric and magnetic fields $\vec{E}$ and $\vec{H}$). These electrons subsequently diffuse the absorbed energy in the interaction volume due to their thermal conductivity. In the absorption region, they also induce a polarization current $\vec{J}$, the dynamics of which can be modeled using the Drude model that sees the resistivity of the bulk in terms of the scattering of fast electrons by the relatively slow immobile phonons in the metal that act like obstructions to the flow. For Ni metal with ω_{pl} and ν plasma and collision frequencies we have

$$\begin{cases} \frac{\partial \vec{E}}{\partial t} = \frac{\nabla \times \vec{H}}{\epsilon_0} - \frac{1}{\epsilon_0}\vec{J} \\ \frac{\partial \vec{H}}{\partial t} = -\frac{\nabla \times \vec{E}}{\mu_0} \\ \frac{\partial \vec{J}}{\partial t} + \vec{J}\nu = \epsilon_0 \omega_{pl}^2 \vec{E} \end{cases} \tag{3.2.2}$$

The rate of energy absorbed by the conduction layer is $I\alpha_{abs}$, where $I = \frac{1}{2}\sqrt{\frac{\epsilon_0}{\mu_0}}\left|\vec{E}\right|^2$ is the intensity and α_{abs} is the absorption coefficient related to the extinction coefficient k and the wavelength

as $\alpha_{abs} = \frac{4\pi k}{\lambda}$.

The two-temperature electron-lattice heat transfer and diffusion phonon dynamics was recently modelled with the compressible Navier-Stokes equations [Rud+19a; Rud+20] as follows:

- **Continuity equation:**

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0, \tag{3.2.3}$$

- **Conservation of energy for electrons:**

$$C_e \frac{\partial T_e}{\partial t} = \nabla \cdot (k_e \nabla T_e) - \gamma_{ep}(T_e - T_p) + I\alpha_{abs}, \tag{3.2.4}$$

- **Conservation of energy for phonons:**

$$\rho C_p \left[\frac{\partial T_p}{\partial t} + \vec{u} \cdot \nabla T_p \right] = \nabla \cdot (k_p \nabla T_p) + \gamma_{ep}(T_e - T_p), \tag{3.2.5}$$

- **Conservation of momentum:**

$$\frac{\partial (\rho \vec{u})}{\partial t} + (\vec{u} \cdot \nabla)(\rho \vec{u}) + (\rho \vec{u}) \nabla \cdot \vec{u} = -\nabla (P_e + P_p) + \mu \nabla^2 \vec{u} + \frac{1}{3} \mu \nabla (\nabla \cdot \vec{u}), \tag{3.2.6}$$

- $\vec{u}$ and ρ are the fluid velocity and density, respectively.
- T_e and T_p denote the electron and lattice temperatures.
- P_e, C_e, and k_e represent the electronic pressure, the electron heat capacity, and conductivity, respectively. These are evaluated based on the results of ab initio calculations [Bév+14].
- P_p is the lattice pressure, defined by the equation of state [LBF94].
- μ, C_p, and k_p correspond to the ion viscosity, heat capacity, and thermal conduction, respectively
- γ_{ep} represents the electron-ion coupling constant or collision frequency, quantifying the rate of energy exchange between electrons and ions.

These equations were solved using a finite difference scheme with a hexagonal distribution of half-spherical nanometer-scale cavities as initial conditions. For the simulations, two cross-polarized pulses, having a combined fluence $F = 0.18$ J/cm^2, were applied with a delay of 8 ps to replicate experimental irradiation conditions.

3.2.4 Dynamics when diffusion dominates exchange

Electrons, due to their much smaller mass compared to ions respond much more quickly to changes in the environment than ions do. We can thus assume that electron dynamics occurs on a much faster timescale than ion dynamics. In the sequel, we shall assume that enough time has passed that $I\alpha_{abs} \approx 0$ (so of the order of the femtosecond in our experimental setting). This energy from the laser that was just absorbed creates a great variation of ion and electronic temperature gradients.

We shall assume that we are at a time scale in which these gradients are still important, but enough time has passed that thermalization, which will end at circa $\tau =\approx 10$ ps, has started to occur. At this time, the term $\gamma_{ep}(T_e - T_p)$ is small compared to both $\nabla \cdot (k_e \nabla T_e)$ and $\nabla \cdot (k_p \nabla T_p)$: diffusion dynamics dominates electron-ion dynamics (which is consistent with a electron blast wave).

This is a *different* time interval than that which is analyzed in [Rud+19a], which examines the dynamics *after* thermalization, and studies the effect of the ionic pressure wave propagation and subsequent rarefaction wave (see Fig.1 in [Rud+19a]). We rather study the *first* pressure wave and subsequent rarefaction wave, which take place early in the dynamics, as can be seen in the leftmost image in Fig. 3.5. The latter induces a pressure gradient in the electronic fluid, that has opposite sign to that of the ionic fluid, and thus acts as buoyancy term in the hydrodynamics equations for ions. This, for a sufficiently wide laser spot and a double-pulse experimental that breaks preferential spacial direction, can be approximately described in terms of the Boussinesq approximation (valid only in the very short time frame up to thermalization) and hence, as shown in the original paper by Swift and Hohenberg [SH77], vertical convective motion known as Rayleigh-Bénard convection, which can be modelled by the Swift-Hohenberg equation.

This convective process, because it takes place at depths of the order of tens of nanometers, is more consistent with the sub-wavelength feature size of circa 50 nm reported in the literature [Nak+21a] – since for Rayleigh-Bénard convection the radius of convection cells should be approximately equal to the depth of the liquid layer, as stated in the last equation of Sec. 3.2.2.

The duration of the process, of the order of 10 ps for Ni as can be observed in Fig. 3.5, is also consistent with the determinant role that inter-pulse delay has on the observed patterns. If the time delay between pulses is large, then the second pulse acts on a layer of a different depth and different material properties. We would expect to observe a change in pattern shape and spatial frequency.

If the second shock wave comes in when the blast wave is in effect, it will result in a weaker second blast wave: we have less electronic buoyancy, and we expect the order parameter to be closer to zero. This should result in less spatial wavelengths to be selected (via the mechanism described in Sec. 3.5.1), and we should expect more ordered, patterns. By inspection of Fig. 3.5, this happens in Ni between 2 and 5 ps. And indeed, 4 ps is he time delay for which the highly-ordered nanopeaks reported in [Nak+21a]) are experimentally observed.

On the other hand, this also suggests that the patterns multi-modality, that is observed *discontinuously* at certain delays, could be explained via a resonance mechanism: only when the second pulse appears at a depth that is a multiple of the first pulses' final depth, will the second feature size not destroy the prior features.

Finally, this new mechanism of pattern formation offers an alternative explanation for the formation of experimentally observed hexagonal patterns, for which a process known as Marangoni convection instability has recently been proposed as the only reasonable candidate [Abo+20] to explain their formation. However, the full picture is more complex, and such fluid flow induced by Rayleigh-Bénard-Marangoni hydrodynamic instability requires isotropic thermal gradients conditions to occur upon ultrashort laser irradiation [Rud+20]. Indeed, below we show that certain hydrodynamic equations (Boussinesq) approximately govern the dynamics after the early stages of laser-matter interactions and before thermalization. This, together with the symmetries imposed by the experimental setup, lead to a Rayleigh-Bénard type convection governed by the SH equation, which indeed cannot produce hexagonal patterns. However, as was shown in [Hak77], the modified SH equation with a quadratic non-linearity can be obtained with the same experimental assumptions by replacing the Boussinesq equations with their compressible flow analog. This second

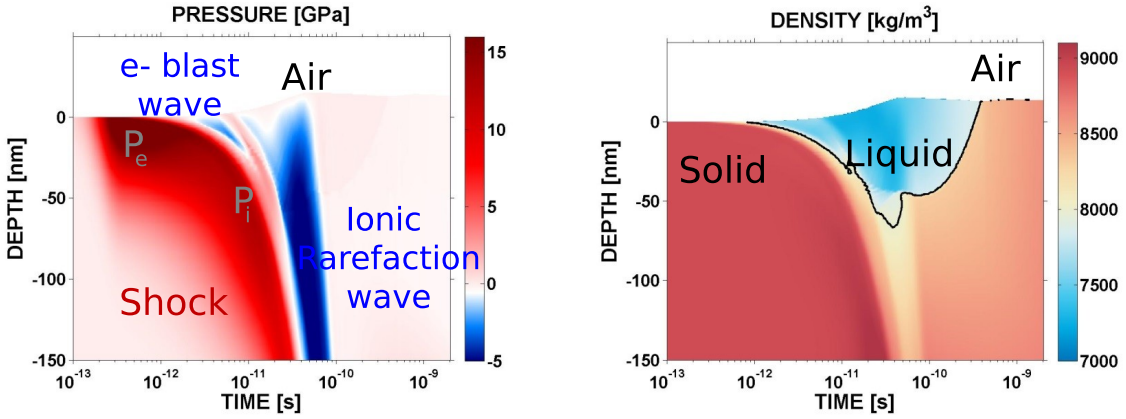


Figure 3.5: Pressure and temperature hydrodynamics of femtosecond laser-Nickel interaction (Jean-Philipe Colembier, private communication). In prior work, emphasis was on $t > 5ps$, the thermalization time, where one can observe a pressure wave and the subsequent rarefaction wave (blue). As can be shown in hydrodynamic simulations, before thermalization time $\approx$ 10ps, a similar event takes place, at a much thinner layer, of up to 50 nm.

equation, which is called *Haken model* in [Wal12] but is more generally referred to as the modified SH model in the literature, *can* produce hexagonal patterns.

Because the convective process is so short-lived, only by accumulation of double-pulse events will we be able to observe pattern formation. The full dynamics thus consists of series of short-time Rayleigh-Bénard convection approximately modelled by the Swift-Hohenberg equation, interrupted by the subsiding of the electronic blast wave and finally fixed by solidification.

3.2.5 Recovering the Boussinesq approximation from the two temperature model

We now show that we can recover the Boussinesq approximation from the two temperature model.

Assume first that $k_e \approx k_0 T_e$ as in [Cor+88] and $\gamma_{ep} \approx c_1 e^{-c_2 T_e}$ as seen in [LZC08], Fig.5(d), where $c_1 \approx 8, c_2 \approx 2\ln 2$ are positive constants[3], and get, for the first condition

$$\begin{aligned} \nabla \cdot (k_e \nabla T_e) \gg \gamma_{ep}(T_e - T_p) \approx \gamma_{ep} T_e \Rightarrow \\ k_0 \|\nabla T_e\|^2 + k_0 T_e \nabla^2 T_e \gg c_1 e^{-c_2 T_e} T_e \approx c_1 (T_e - c_2) \Rightarrow \\ k_0 \|\nabla T_e\|^2 + k_0 T_e \nabla^2 T_e \gg c_1 T_e \Rightarrow \end{aligned}$$

which should hold in the case of large temperature gradients.

In the following, we shall make the approximation that k_e, k_p, γ_{ep} are spatially constant with respect to the gradient of the respective temperatures. With that approximation we obtain

$$\frac{\|\nabla^2 T_e\|}{|T_p - T_e|} \gg \frac{\gamma_{ep}}{k_e}$$

[3] c_2 has units of inverse temperature and c_1 units of γ_{ep}.

and similarly for the ions.

, the electronic conservation of energy equation (3.2.4) becomes

$$C_e \frac{\partial T_e}{\partial t} = \nabla \cdot (k_e \nabla T_e) - \gamma_{ep}(T_e - T_p) + I\alpha_{abs} \approx k_e \nabla^2 T_e$$

To investigate this equation further, assume that the interaction domain is a thin volume that extends infinitely in the the x- and y- directions. Then T_e will only vary in the z direction and all spatial derivatives in the x and y direction will be zero. This allows us to rewrite the electronic conservation of energy (3.2.4) above as

$$\frac{\partial T_e}{\partial t} = \frac{k_e}{C_e} \frac{\partial^2 T_e}{\partial z^2}$$

This is a wave equation, and the particular form of its solution depends on boundary conditions. At the time t_0 of the beginning of this evolution, the laser energy has been absorbed by the conduction band and has had some time to equalize. The electronic temperature thus decreases with depth from $T_e(0, t_0)$ exponentially, that is $T_e = -a\frac{\partial T_e}{\partial z}$, where $a > 0$ is a constant. Ths defines how abruptly the electronic temperature decays from its maximum at time t_0, which depends on the amount of energy that was deposed, etc. We assume a separable solution $T_e(z, t) = g(z)h(t)$. Substituting this above yields

$$g(z)\frac{d^2h}{dt^2} = \frac{k_e}{C_e} h(t) \frac{d^2g}{dz^2}.$$

Dividing both sides by $\frac{k_e}{C_e} g(z)h(t)$, we get

$$\frac{1}{\frac{k_e}{C_e} h(t)} \frac{d^2h}{dt^2} = \frac{1}{g(z)} \frac{d^2g}{dz^2}.$$

Since the left-hand side is only a function of t and the right-hand side is only a function of z, both sides must be equal to a constant, say $-\omega^2$:

$$\frac{d^2h}{dt^2} + \omega^2 h(t) = 0, \quad \frac{d^2g}{dz^2} + \omega^2 \frac{k_e}{C_e} g(z) = 0.$$

We also assume that $\frac{\partial T_e}{\partial z} = aT_e$, or $g'(z) = a \cdot g(z)$, decreasing exponentially from the bottom $z = h$ up to $z = 0$ where $g(0) = U$, which is compatible with rarefaction. Solving for $g(z)$, we get

$$g(z) = Ae^{az}.$$

Using the condition $g(0) = U$, we find $g(z) = Ue^{az}$. Plugging this back into the wave equation for g, we obtain $\omega^2 = -a^2 \frac{C_e}{k_e}$ which implies that ω is imaginary. The general solution for $h(t)$ can thus be written as

$$h(t) = Ce^{\sqrt{|\omega|}t} + De^{-\sqrt{|\omega|}t}.$$

Combining these, the full solution becomes

$$T_e(z, t) = Ue^{-az}\left(Ce^{\sqrt{|\omega|}t} + De^{-\sqrt{|\omega|}t}\right).$$

Finally, we use the boundary conditions $h(0) = 1$ and $T_e(z, \tau) = U$, where τ is the equalization time (of the order of 10 ps for Ni). The condition $h(0) = 1$ can be satisfied with $C + D = 1$, but since $T_e(z, \tau) = U$ we get

$$C = \frac{1 + e^{\sqrt{|\omega|}\tau}}{2}$$
$$D = \frac{1 - e^{\sqrt{|\omega|}\tau}}{2}$$

which, together with $\sqrt{|\omega|} = a\sqrt{\frac{C_e}{k_e}}$ yields the following solution for the electronic temperature for $t_0 < t < \tau$:

$$T_e(z, t) = Ue^{az}\left(Ce^{a\sqrt{\frac{C_e}{k_e}}t} + De^{-a\sqrt{\frac{C_e}{k_e}}t}\right).$$

Let us now examine the phonon conservation of energy equation. Requiring that the fluid be incompressible we get

$$\begin{aligned}\frac{\partial T_p}{\partial t} + \vec{u} \cdot \nabla T_p &= \frac{1}{\rho C_p}\nabla \cdot (k_p \nabla T_p) + \frac{\gamma_{ep}}{\rho C_p}(T_e - T_p) \\ &= \frac{1}{\rho C_p}\nabla \cdot (k_p \nabla T_p)\end{aligned}$$

where in the last step we used the hypothesis that for $t_0 < t < 5$ ps the diffusive dynamics dominates. If k_p is a spatial constant, the phonon energy conservation equation further simplifies

$$\frac{\partial T_p}{\partial t} + \vec{u} \cdot \nabla T_p = \frac{k_p}{\rho C_p}\nabla^2 T_p.$$

We now show that the phonon fluid follows the Boussinesq approximation. Typically in this setting, we assume that fluid density is first-order constant $\rho = \rho_0 + \delta\rho$ with $|\delta\rho| \ll |\rho_0|$, which is sensible for phonons. The the continuity equation becomes

$$\nabla \cdot \vec{u} = 0,$$

Replacing the new continuity equation in the momentum conservation equation we get

$$\begin{aligned}\frac{\partial(\rho\vec{u})}{\partial t} &= -(\vec{u} \cdot \nabla)(\rho\vec{u}) - (\rho\vec{u})\nabla \cdot \vec{u} - \nabla(P_e + P_p) + \mu\nabla^2\vec{u} + \frac{1}{3}\mu\nabla(\nabla \cdot \vec{u}) \\ &= -(\vec{u} \cdot \nabla)(\rho\vec{u}) - \nabla(P_e + P_p) + \mu\nabla^2\vec{u}\end{aligned}$$

since $\frac{\partial(\rho_0\vec{u}+\delta\rho\vec{u})}{\partial t} = \rho_0\frac{\partial\vec{u}}{\partial t} + \vec{u}\frac{\partial(\delta\rho)}{\partial t} \approx \rho_0\frac{\partial\vec{u}}{\partial t}$ and since, for the convective term we have

$$\begin{aligned}(\vec{u} \cdot \nabla)((\rho_0 + \delta\rho)\vec{u}) &= (\vec{u} \cdot \nabla)(\rho_0\vec{u}) + (\vec{u} \cdot \nabla)(\delta\rho\vec{u}) \\ &\approx (\vec{u} \cdot \nabla)(\rho_0\vec{u}) \\ &= \rho_0(\vec{u} \cdot \nabla)\vec{u},\end{aligned}$$

(where we assumed small velocities and spacial gradients of $\vec{u}$ and $\delta\rho$, and used the fact that ρ_0 is constant in space[4]), the modified momentum conservation equation will thus be:

$$\rho_0 \frac{\partial \vec{u}}{\partial t} = -\nabla P_p + \mu \nabla^2 \vec{u} - \rho_0 (\vec{u} \cdot \nabla)\vec{u} - \nabla P_e$$

where we collected all the terms other than the inertial term on the right-hand side. This equation is analogous to the one we obtain in the classical Boussinesq approximation with the gravity-based buoyancy replaced by the negative gradient of electronic pressure $-\nabla P_e$, which for the duration of the blast wave, has the opposite sign of that of $-\nabla P_p$.

Assuming that we are sufficiently close to equilibrium that P_e is proportional to T_e (which should hold locally), then $-\nabla P_e = -C\nabla T_e$ (in the ideal gas limit, the constant would be the Boltzmann constant). Using the expression we found above for T_e, expanding around $z = h$ and keeping only up to first-order terms we have

$$\begin{aligned} T_e(z,t) &\approx Uh(t) + azUh(t) \Rightarrow \\ \nabla T_e &\approx aUh(t)\vec{e}_z \approx aT_p\vec{e}_z \end{aligned}$$

where $\vec{e}_z$ points down. Replacing this in the expression above we obtain

$$\begin{aligned} \rho_0 \frac{\partial \vec{u}}{\partial t} &= -\nabla P_p + \mu \nabla^2 \vec{u} - \rho_0 (\vec{u} \cdot \nabla)\vec{u} - C\nabla T_e \\ &= -\nabla P_p + \mu \nabla^2 \vec{u} - \rho_0 (\vec{u} \cdot \nabla)\vec{u} - CaT_p\vec{e}_z \end{aligned}$$

In summary, we have found, for the ion temperature and velocity:

$$\nabla \cdot \vec{u} = 0 \tag{3.2.7}$$

$$\frac{\partial T_p}{\partial t} + \vec{u} \cdot \nabla T_p = \frac{k_p}{\rho C_p} \nabla^2 T_p \tag{3.2.8}$$

$$\rho_0 \frac{\partial \vec{u}}{\partial t} = -\nabla P_p + \mu \nabla^2 \vec{u} - \rho_0 (\vec{u} \cdot \nabla)\vec{u} \underbrace{-aCT_p\vec{e}_z}_{\text{e}^- \text{ buoyancy}}, \tag{3.2.9}$$

which are the Boussinesq equations for fluid motion, with a modified "electronic buoyancy" originating from the electronic pressure gradient term from the blast wave, which is proportional to $-\vec{e}_z$ and points up.

Summary As shown in Fig. 3.3, after a short period of laser-matter interaction, matter is left to evolve, a process which can modelled using eqs. 3.2.3, 3.2.4, 3.2.5, and 3.2.6. As we showed above, this can be seen, at the time scale of pre-electron temperature equalization, as ion fluid motion using the Boussinesq approximation, with the gravitational buoyancy term being replaced by an *electronic* buoyancy originating by the blast wave near the surface of the solid.

This, as shown in [SH77] generates a shortly lived Rayleigh-Bénard type convection of the ion fluid, the unstable 2D-dynamics of which can be described using what is known as the Swift-Hohenberg equation (cf. Sec. 3.5.2).

In between pulses the electron blast wave subsides, convection stops, matter becomes solid again, and all temperatures equalize. The process begins again at each new impulsion (possibly

[4]Nothing much happened on the phonon side yet, so this should be okay?

interacting with previous blast waves), following an approximately SH-dynamics at the scale of the time-difference between double pulses.

This physical model cannot be expected to explain the full dynamics of pattern formation. Before we proceed, we shall therefore examine the problem of learning with data by integrating *partial* physical information.

3.3 Learning by integrating partial physical information: related work

Given a physical phenomenon modeled by a PDE, the problem of predicting the result of measurements is known as the *forward problem*, while estimating unobserved states and parameters that characterize the system — which are needed to solve the forward problem — is called the *inverse problem*. The solution of the forward problem for deterministic PDEs is generally unique, but that of the inverse problem is not. It is rather naturally expressed as a probability distribution, which motivates a rigorous formulation of the inverse problem in terms of Bayesian inference, with well-established methods going back to Laplace [Tar05].

Incorporating prior knowledge and combining it with data is the key problem in Bayesian estimation. Since priors and data are domain specific, it should come as no surprise that methods to tackle the inverse problem have been developed in parallel in several domains where physical knowledge can be expressed in terms of a PDE and data is abundant.

In geophysics and climate science, physical models are sophisticated and well-established but there is only partial information about state: satellite data, for example, is given in patches, but forecasting using the physical model requires knowledge of the full state; solving this inverse problem in this domain is commonly done using a collection of methods known as *data assimilation* [Car+18].

In the physics community, on the other hand, one can often carefully prepare experiments to set the initial state, and the main goal is now physical model development or validation. When the initial state is known, the inverse problem of finding the distribution of model parameters is thus the main focus. It is known as *model calibration* [KO01], and a host of recent results exist using machine learning techniques, notably deep learning [VS21], to learn the parameters of either the full model[5] or a correction to incomplete physical knowledge [Yin+21] (an *augmented* model) from data.

More specifically in photonics, the focus is on inverse design [Mol+18] — algorithmic techniques for discovering optical structures based on desired functional characteristics —, with neural networks being used to speed up optimization of nano-photonics structures, by replacing the forward model with a much faster to evaluate neural network surrogate [Wie+21; Ma+21], for example. The task is reminiscent of our own. Unlike in our case, however, the physical model is assumed to be complete, the system state can be prepared, and data is abundant. Crucially, there is a a specific functional characteristic to optimize for, whereas we are interested in exploration rather than design.

In reality there is always some uncertainty with respect to either model parameters or system state. To solve the *joint* inverse problem of finding state and model parameters, several approaches involving machine learning were recently proposed in the climate sciences. The main idea is to alternate a data assimilation step to estimate state and a machine learning step to learn model parameters from data [Fil+20; Far+21; Ngu+20; Bra+21]. Because in the climate science case the

[5] Which is typically carefully constrained as to incorporate the right biases, and can thus be seen as already incorporating physical knowledge.

dynamics are generally well-known, the models that are jointly learned can be seen as *corrections* to partial physical knowledge given as a governing PDE, and the goal is to learn them from data (although the correction itself also commonly incorporates physical knowledge or symmetries in the form of constraints imposed during training [Déc+20; GDY19; Che+19; Cra+20; Don+22] and/or in the network architecture [Beu+19; Fil+20]).

In these terms our task of predicting new laser patterns can be framed as follows: we have a complex system in which the laser-matter interaction takes place; X_t, the state of this system at time t is unknown but for its noisy image through an observation operator $Y_t = \mathcal{H}X_t + \epsilon$, which consists in the laser parameters and the SEM images of the material at the zone of incidence of the laser spot after solidification (the solidification time is not fixed and depends, among other things, on laser fluence). As in the climate science approaches above, we have an explicit partial model for the transition from state X_t to X_{t+1}, the Swift-Hohenberg equation with unknown parameters φ, which we can see as a first order model of the physical phenomenon[6], and we have no access to the hidden state. Crucially, unlike these approaches, we have a single observation for each evolution and several orders of magnitude less evolution data. This makes the dual inverse problem of inferring initial state and model parameters unfeasible, and the aforementioned methods inapplicable.

We shall overcome this difficulty by recasting the problem in such a way that for our purposes, in the case of self-organization pattern forming dynamics, the state determination inverse problem is unnecessary, in the sense that a feature transformation F exists such that in the image of F, patterns can be effectively described via model parameters alone[7]. This turns the high-dimensional ill-posed inverse problem into a much lower-dimensional problem of learning the relationship between laser parameters and model parameters using data.

3.4 Integrating partial physical information to learn with few data and no knowledge of initial conditions

In this section, we derive a principled approach to the problem of learning a complex relationship between an observable quantity θ and a physical field u, given only few data, by taking advantage of partial contextual physical knowledge in the form of a differential equation on u not explicitly depending on θ, in the absence of knowledge of initial conditions. We strive for generality but we shall refer to the main subject of this chapter for examples and clarification.

This problem is fundamentally difficult in three interacting ways: *first*, the dearth of experimental data makes learning a complex relationship directly, from data alone, impossible. This is usually where a physical model can be of assistance, typically via data augmentation: one uses a differential equation solver — or a neural network surrogate of a differential equation solver — to generate more θ, u pairs in a principled, physically consistent manner, which we can then use to complement the small dataset size. Even if the physical information is not complete, it can still be of use: if the model is only approximate, one splits the field in two components, one which is exactly modeled by the physical model, and a second which can be seen as a perturbation of the former, and which can be learned from data [Yin+21]. But producing a model explicit on θ is the *second* difficulty, particularly in the earlier stages of the experimental process or in the case where θ are difficult to interpret in terms of known physical quantities commonly used to parameterize the

[6] In our case, we are solving the inverse problem for the first order model, not a correction.

[7] This feature transformation is shown to exist in Section 3.4.4 for the case of models with pattern-like solutions (which is our case, as detailed below in Section 3.5.1)

physical model. This is not an insurmountable difficulty: we may lack explicit physical knowledge, but it is rarely the case that we approach a physical situation without any sort of background knowledge. In most situations, a *general* model relating common physical observables and the physical field can be produced, derived on first principles or symmetry arguments. This can be a considerable simplification, as it may allow us to turn the problem of learning the relationship between the θ and u into one of learning the relationship between the θ and the general model parameters φ. As the dimension of φ is generally lower than the dimension of u, it is much easier to learn the latter. Herein lies the *third* difficulty: if the general model is good and the observables are sufficiently informative, one can hope that the number of data required for calibration be small. But calibration data is of a different nature as it is typically done via a series of carefully designed experiments [CH93]. Learning the parameters of a differential equation in time typically involves knowledge of u at different times, which is not necessarily part of the original θ, u data — we need then knowledge of initial conditions for u.

We propose to address these difficulties as follows: we first show that in the case of self-organization, pattern-forming processes, the information contained in the initial conditions u_0 is *small*, in the sense that a non-injective feature transformation F exists such that they can be losslessly expressed in the lower-dimensional $F(u_0)$, which makes the dependency on initial conditions less important. We then proceed by making two assumptions on the physical process and its relationship with the observables, allowing us to propose a method to take advantage of self-organization physical knowledge to learn with few data. We note that these assumptions hold in a number of interesting cases, namely climate models, where our method could be also applied to find the relationship between observed quantities and features of rare events, for example (extreme event prediction, of which, by definition, we have few data).

3.4.1 Problem statement

Consider a physical field $u(x,t) := u$ which we believe is mainly the result of a certain physical process, the evolution of which can be described in terms of physical process parameters φ and initial conditions u_0 (compatible with the physical situation) and by a PDE $\dot{u} = f(\varphi, u_0)$[8]. Assume we have no knowledge of either φ or u_0 — although the latter is assumed to belong to a large set $\mathcal{U}_0$ of "*reasonable*" initial conditions compatible with the physical situation. We also assume that a certain *observed* quantity θ (which in our case will correspond to the laser parameters) exists which affects φ — although it may also affect other unknown latent variables, which may in turn affect u (Fig. 3.6, left).

We would like to sample from the distribution of u given a certain value of the observed quantity θ. Unfortunately, since we have no knowledge of the initial conditions, $p(u|\theta)$ cannot be calculated directly

$$\begin{aligned} p(u|\theta) &= \sum_{u_0} \sum_{\varphi} p(u, u_0, \varphi|\theta) \\ &= \sum_{u_0} \sum_{\varphi} p(u|u_0, \theta, \varphi) p(u_0, \varphi|\theta), \end{aligned}$$

[8]Throughout this section we assume that u is a real field and periodic boundary conditions; we further assume that conditions are satisfied such that u is unique (existence is posited since we are assuming that the process is modeled by this equation and we observe the fields)

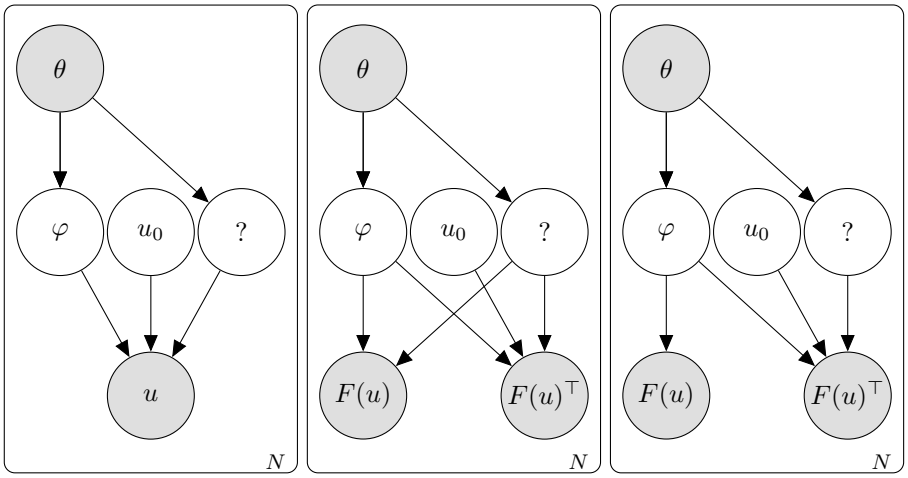


Figure 3.6: Left: Observable θ influences a physical process φ, among others (marked '?'), which, together with unobserved initial conditions u_0, determine observed field u. Center: A feature transformation exists such that $u = (F(u), F(u)^{\perp})$, and the observable $F(u)$ is useful and independent of initial conditions u_0. Right: in addition, the feature transformed observable is not affected by the other physical processes.

(where sums are to be replaced by integrals in the continuous case) and we shall need to make hypothesis about the physical process to proceed.

3.4.2 Two hypotheses on the physical process

The distribution $p(u|\theta)$ may be impossible to calculate directly, but the quantity of interest is often not u, but rather a certain function of u, $F(u)$, where $F(u)$ is typically simpler than u, which motivates the following hypotheses:

Hypothesis 3.1. *A function F of the field u exists such that the knowledge of the physical process φ suffices to determine the likelihood of $F(u)$ conditioned on θ. Precisely, $p(F(u)|u_0, \theta, \varphi) = p(F(u)|\theta, \varphi)$. We call* feature space *the image of F on the domain of u.*

Under Hypothesis 3.1, and assuming that the initial conditions are independent of the physical process φ and the observed quantity θ (as shown in Fig. 3.6), we have

$$\begin{aligned} p(F(u)|\theta) &= \sum_{u_0}\sum_{\varphi} p(F(u)|u_0, \theta, \varphi)p(u_0, \varphi|\theta) \\ &= \sum_{u_0}\sum_{\varphi} p(F(u)|\theta, \varphi)p(u_0|\varphi, \theta)p(\varphi|\theta) \\ &= \sum_{u_0}\sum_{\varphi} p(F(u)|\theta, \varphi)p(u_0)p(\varphi|\theta) \\ &= \sum_{\varphi} p(F(u)|\theta, \varphi)p(\varphi|\theta). \end{aligned}$$

To proceed, we make the following additional assumption:

Hypothesis 3.2. *The observed quantity θ determines the physical process: $p(\varphi|\theta)$ is peaky, in the sense that there is a $\tilde{\varphi}$ such that $p(\varphi|\theta) = 0$ for $\varphi \neq \tilde{\varphi}$. In particular, if p is continuous, $p(\varphi|\theta) = \delta(\varphi - \tilde{\varphi}(\theta))$, where $\delta(\cdot)$ refers to the peaky distribution related to physical process.*

Under this assumption, $\tilde{\varphi}$ is a function of θ, which implies that θ is at least locally a function of $\tilde{\varphi}$ and we can write

$$\begin{aligned} p(F(u)|\theta) &= \sum_{\varphi} p(F(u)|\theta, \varphi) p(\varphi|\theta) \\ &= p(F(u)|\theta(\tilde{\varphi}), \tilde{\varphi}) p(\tilde{\varphi}|\theta) \\ &= p(F(u)|\theta(\tilde{\varphi}), \tilde{\varphi}) \\ &= p(F(u)|\tilde{\varphi}). \end{aligned}$$

How can we find $\tilde{\varphi}(\theta)$?

3.4.3 Learning by maximizing the likelihood

Our general method to learn $\tilde{\varphi}(\theta)$ is to maximize the likelihood of the observations. In the sequel, we propose two strategies to do so: one in which we maximize it directly, and another in which we maximize a lower bound.

Maximizing the likelihood directly

Having access to experimental data $\left\{\theta^i, u^i\right\}_{i=1\dots N}$, we can parameterize $\tilde{\varphi}$ with a Neural Network (α), for example, and maximize the log likelihood of observing the $F(u^i)$:

$$\bar{\alpha} = \arg\max \sum_{i=1}^{N} \log p\left(F(u^i)|\tilde{\varphi}(\theta^i); \alpha\right). \tag{3.4.1}$$

If u is high-dimensional, the relationship between $F(u)$ and $\tilde{\varphi}$ is potentially complex, and thus requires N large to model satisfactorily. Having physical knowledge in the form of a differential equation solver $u = \text{Solver}(\tilde{\varphi}, u_0)$, however, considerably simplifies the problem.

In feature space, for a process respecting Hypothesis 3.1, we can choose arbitrary initial conditions and fit only the relationship between θ and the *parameters* of the differential equation, which generally have much lower dimension than the field u. Since this is a much simpler task, we expect that a much smaller N will suffice to produce a satisfactory model.

Assuming data is generated i.i.d. from a fixed-variance Gaussian distribution, this corresponds to minimizing the mean squared error between the images, through F, of experimental fields, and fields generated by the PDE solver for an arbitrary initial condition $u_0 \in \mathcal{U}_0$.

$$\bar{\alpha} = \arg\min \frac{1}{N} \sum_{i=1}^{N} \left\| F(u^i) - F(\text{Solver}(\tilde{\varphi}_\alpha(\theta^i), u_0)) \right\|^2 \tag{3.4.2}$$

Note that this optimization is more conveniently done with a differentiable *surrogate* of the solver.

Maximizing a lower bound of the likelihood

Note that[9]

$$\begin{aligned}\log\left(p(F(u)|\theta(\tilde{\varphi}),\tilde{\varphi})p(\tilde{\varphi}|\theta)\right) &= \log p(F(u)|\theta(\tilde{\varphi}),\tilde{\varphi}) + \log p(\tilde{\varphi}|\theta)\\ &= \log p(F(u)|\tilde{\varphi}) + \log p(\tilde{\varphi}|\theta).\end{aligned}$$

Let $\tilde{\varphi}_1$ be the maximizer of the first term, $\tilde{\varphi}_1 = \arg\max_{\tilde{\varphi}} \log p(F(u)|\tilde{\varphi})$. Then

$$\max_{\tilde{\varphi}} \left\{\log p(F(u)|\tilde{\varphi}) + \log p(\tilde{\varphi}|\theta)\right\} \geq \log p(F(u)|\tilde{\varphi}_1) + \log p(\tilde{\varphi}_1|\theta). \tag{3.4.3}$$

Provided we find $\tilde{\varphi}_1^i$ for each u^i, we can replace the log likelihood maximization objective with that of maximizing a lower bound:

$$\bar{\alpha} = \arg\max_{\alpha} \sum_{i=1}^{N} \log p\left(\tilde{\varphi}^i|\theta^i;\alpha\right) \tag{3.4.4}$$

which, repeating the argument above, corresponds to minimizing the mean squared error,

$$\bar{\alpha} = \arg\min_{\alpha} \frac{1}{N}\sum_{i=1}^{N} \left\|\tilde{\varphi}^i - \tilde{\varphi}_\alpha(\theta^i)\right\|^2.$$

We can replace this task with maximizing a lower bound, *after* having found the first maximizer. It remains to find the $\tilde{\varphi}^i$. To do so, we note that having an efficient solver and assuming sufficient regularity, one can pre-generate sufficiently many fields $\mathcal{U}_g = \left\{u_g^k\right\}_{k=1\ldots M}$ such that the expected distance to the nearest neighbor in feature space, given by F, is as small as one would like:

$$\begin{aligned}\delta &= \frac{1}{M}\sum_{k=1}^{M} \min_{j\neq i} \left\|F(u_g^k) - F(u_g^j)\right\|^2\\ &= \frac{1}{M}\sum_{k=1}^{M} \left\|F(u_g^k) - F(\hat{u}_g^k)\right\|^2,\end{aligned}$$

where we denoted $\hat{u}_g^k$ as the nearest neighbor in $\mathcal{U}_g$ of u_g^k, in feature space. Denoting $\text{Solver}(\tilde{\varphi}_\alpha, u_0) := u^\alpha$ for simplicity and $\hat{u}^\alpha$ its nearest neighbor in $\mathcal{U}_g$, we have

[9] We recall that even though we are under the conditions of Hypothesis 3.2 — φ is a function of θ —, we don't know which function that is, which explains keeping around the term $\log p(\tilde{\varphi}|\theta)$, which we will maximize for experimental data in order to find $\tilde{\varphi}$.

$$\begin{aligned}\min_\alpha \frac{1}{N}\sum_{i=1}^N \left\|F(u^i)-F(u^\alpha)\right\|^2 &= \frac{1}{N}\sum_{i=1}^N \min_\alpha \left\|F(u^i)-F(\hat{u}_p^i)+F(\hat{u}_p^i)-F(u^\alpha)\right\|^2 \\ &\leq \frac{1}{N}\sum_{i=1}^N \left\|F(u^i)-F(\hat{u}_p^i)\right\|^2 + \min_\alpha \frac{1}{N}\sum_{i=1}^N \left\|F(\hat{u}_p^i)-F(u^\alpha)\right\|^2 \\ &\leq \frac{1}{N}\sum_{i=1}^N \left\|F(u^i)-F(\hat{u}_p^i)\right\|^2 + \delta \\ &\leq 2\delta.\end{aligned}$$

Since $\delta \to 0$ we can set $\tilde{\varphi}^i$ as the solver parameter of the nearest neighbor in feature space, among pre-generated fields, of u^i.

3.4.4 Generality of the two hypotheses

The hypotheses above correspond to the following desiderata:

- A useful, simplifying and separating feature space F exists.
- In F, the task is independent of initial conditions.
- In F, the physical process φ is essentially a function of the observed θ.

which we now examine in turn.

Choosing a useful, simplifying, separating feature space

In the exploratory stages of research, an explicit measure of usefulness is typically not available, because future applications are unknown. It is thus important to keep the feature transformation F as discriminating as possible. Without imposing further constraints, this is trivially satisfied by choosing a bijection. On the other hand, we want F to be simplifying, in the sense that it is invariant to quantities which we do not care for. Without further constraints, this objective is satisfied trivially by choosing F as projection to a single point. These two objectives, which we illustrate in Fig. 3.7, are in contradiction and fulfilling them simultaneously is not trivial. This strategy could still be pursued in principle by training a model for a set of broadly defined constraints, but doing so is expensive in terms of time and data, and cannot be justified in practice in the early stages of research. The problem is reminiscent of that in [Beu+21], in which F is called a *physical rescaling*; as in our work, the assumption is that it is inconvenient to train a model to find such a rescaling, which is thus obtained based on physical knowledge and/or statistical properties leaving the target variables invariant. This method is interesting but, in a setting of partial physical information and few data, explicitly defining discriminating invariant feature transformations based on physical or statistical arguments is challenging.

A reasonable alternative in this setting is to choose a simplifying feature transformation F which, applied to similar data, is known to allow a broad task to be performed, which is the strategy that we propose. This has the inconvenience of the feature transformation F not being uniquely defined, which is why we now propose a method to compare several such transformations.

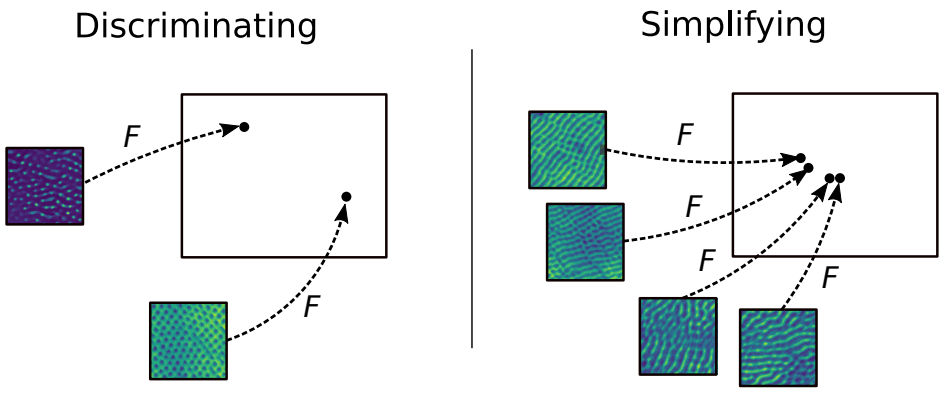


Figure 3.7: One would like the feature transformation of choice F to be simplifying in the sense that it abstracts away pattern minutia while still being discriminating in the sense that patterns which are different will be mapped to different points.

For concreteness and for the sake of clarity, we present this strategy in the context of our specific task of finding a feature space for predicting novel laser-induced nanoscale patterns (shape and scale). The shape of the patterns observed in SEM images is described by the experts in somewhat loose terms such as 'labyrinthine', 'hexagons', 'bumps', 'peaks'. Different patterns are believed to have different applications with different utility. However, this strategy can in principle be generally used in the context of the early stages of the experimental process, with few data and partial knowledge, and where a specific measure of utility is not yet established.

A quality measure for the Feature transformation We *define* as a measure of quality of a feature transformation F with respect to physically relevant patterns, the accuracy of the classification task where the patterns are classified as the modal pattern in the clusters obtained by k-means in feature space. To see why, consider $u_i^{(p_1)}, u_j^{(p_1)}$ two fields with the same pattern. If F is *invariant* with respect to this pattern, then $F(u_i^{(p_1)}) = F(u_j^{(p_1)})$. If F is *discriminating*, for all patterns p_k, if $k \neq l$ then for all i, j, $F(u_i^{(p_k)}) \neq F(u_j^{(p_l)})$. It follows that if F is invariant and discriminating, it solves the task of clustering fields according to patterns. This motivates our definition.

In practice, as mentioned in [Beu+21], a fully invariant non-trivial mapping is difficult or impossible to find, which leads us to define invariance with a given tolerance. This motivates our measure of invariance as the accuracy of a distance-based clustering algorithm, which we take as the k-means modal classifier for simplicity.

We note that identification of the modal pattern and choice of the number of clusters k require physical knowledge, and were therefore performed by an expert. On the other hand, this knowledge does not have to be explicitly defined, which allows our method to be applied in the context of partial physical knowledge, in the early stages of the research process. Finally, we also note that the choice of invariant, discriminating feature F transformation is not unique: if F is such a transformation, we can obtain another one by adding a constant feature, for example. All things being equal, we would rather choose an F as simple as possible.

Independence of initial conditions

In general we cannot guarantee that a feature transformation exists that allows abstracting away initial conditions, but we now show that this is the case for pattern-forming systems via a deterministic process. To ground intuition, we recall that patterns are often formed because Fourier modes are selectively amplified/attenuated during the dynamics, the set of frequencies that are *not* driven to zero laying on a band around some critical frequency[10]. To see that necessarily implies that the initial conditions are redundant, we begin by establishing notation and a few definitions: Let u be a real field defined on a finite interval of length L with periodic boundary conditions[11]. Setting $L = 2\pi$ for simplicity, in the frequency domain, the maximum wave number compatible with such boundary conditions is $2\pi/L = 1$. The possible wave numbers are thus $\frac{2\pi}{L}n = n$ for integers $n \geq 1$, and we can write the field as $u = \sum_{n=1}^{\infty} a_n f_n$ where f_n is the Fourier mode with frequency n and $a_n \in \mathbb{C}$ its amplitude.

To each field, we can associate a distribution of its amplitude among the modes by setting $p_n = \frac{|a_n|^2}{\sum_{n=1}^{\infty} |a_n|^2}$.

Finally, to this distribution we can associate the following Shannon Entropy $H(u)$:

$$H(u) = -\sum_{n=1}^{\infty} p_n \log p_n \tag{3.4.5}$$

The general idea of the proof is the following:

1. Let u be a real field on a bounded domain Ω at a fixed time t, which is the result of a deterministic physical process for given initial conditions i. If that is the case, then there is a function $L_t : \Omega \to \Omega$ such that $u = L_t(i)$. The mutual information between the field u at time t and its initial value i is thus $\mathcal{I}(u, i) = \mathcal{H}(u) - \mathcal{H}(u|i) = \mathcal{H}(u)$, since $u = L_t(i)$ and $\mathcal{H}(L_t(i)|i) = 0$.

2. We prove that for general initial conditions the entropy for a self-organization physical process is decreasing $H(u) < H(i)$.

3. If L_t were a bijection, an inverse L_t^{-1} would exist and $I(u, i) = \mathcal{I}(L_t^{-1}(i), i) = H(i)$ by the same argument as above, which would imply $H(i) = H(u)$. Since this is not the case, L_t must map to a lower-dimensional space.

4. In this lower dimensional space, the initial conditions completely determine the field u (identically, since $u = L_t(i)$).

5. Hence, for such a physical process, there is necessarily redundant information in the specification of initial conditions.

It remains to show the following proposition:

Proposition 3.1. *The entropy of a self-organization process is decreasing for general initial conditions.*

[10] There could be more than one such critical frequency

[11] The extension to polytopes is straightforward.

Proof. Consider now a variation in the distribution of amplitudes $(\dot{p}_1, \cdots, \dot{p}_k, \cdots) := \dot{\mathbf{p}}$, where the dot denotes a time derivative. The corresponding time derivative of the entropy is then:

$$\dot{H} = -\sum_{n=1}^{\infty} \dot{p}_n (\log p_n + 1) \tag{3.4.6}$$

$$:= -\dot{\mathbf{p}} \cdot (\log \mathbf{p} + \mathbf{1}), \tag{3.4.7}$$

The maximal change in entropy will be for $\dot{\mathbf{p}}$ aligned with $\log \mathbf{p} + \mathbf{1}$, which is clearly not generally the case for some initial distribution of amplitudes $\mathbf{p}$. For self-organization physical processes, patterns are often formed because a certain range of nodes in the neighborhood of a critical wave number n_0 is amplified, all others being attenuated. One can generally describe this evolution[12] in terms of the amplitude of each node:

$$a_n(t) = e^{-(n-n_0)^2 t} a_n(0) \tag{3.4.8}$$

where we dropped the absolute value for notation simplicity. Assuming that there is no large power mode (all $p_k < 1/2$, which is indeed the case for $k > 2$ for the maxent distribution of Fourier power spectrum, which is the uniform distribution), all components of the vector $\log \mathbf{p} + \mathbf{1}$ are negative. Further assuming that the physical process has the property that the l_2 norm of the field is constant, $\sum_{n=1}^{\infty} |a_n(t)|^2 := 1/\alpha$ we have, by replacing (3.4.8) in the time derivative of the entropy above, that the entropy of the self-organization process H_{so} decreases for every initial distribution of amplitudes:

$$\begin{aligned}
\dot{H}_{\text{so}} &= -\sum_{n=1}^{\infty} \dot{p}_n (\log p_n + 1) \\
&= -\alpha \sum_{n=1}^{\infty} (\dot{a_n^2})(\log p_n + 1) \\
&= -2\alpha \sum_{n=1}^{\infty} \dot{a}_n a_n (\log p_n + 1) \\
&= \alpha \sum_{n=1}^{\infty} (n - n_0)^2 a_n^2 (\log p_n + 1) < 0,
\end{aligned}$$

which proves the claim. As we mentioned above, this decrease is not maximal as it is not necessarily so that to low power modes there will correspond a greater decrease in amplitude. □

The likelihood is peaky: φ is a function of θ

We cannot control whether or not the likelihood $p(\varphi|\theta)$ is peaky. If it is not, then there are several φ that can be associated to the same θ and the log-likelihood is

$$\log p(F(u)|\theta) = \log \sum_{\varphi} p(F(u)|\theta, \varphi) p(\varphi|\theta).$$

[12] These statements are strictly true for a Type 1S instability, near onset, and for a small perturbation of the $u = 0$ solution in the linear approximation, see [CH93] for details.

We can proceed to maximize the likelihood of the data as before

$$\sum_{i=1}^{N} \log p(F(u^i)|\theta^i) = \sum_{i=1}^{N} \log \sum_{\varphi} p(F(u^i)|\theta^i, \varphi) p(\varphi|\theta^i),$$

but in this case, Hypothesis 3.2 does not apply and so the expression does not simplify. If that is the case, we propose a generalization of Hypothesis 3.2 which may still hold.

Likelihood is peaky in different subsets of data If we can split the *data* in subsets[13] $\mathcal{D}_1 = \left\{(u^i, \theta^i)\right\}_{i=1\ldots N-P}$ and $\mathcal{D}_2 = \left\{(u^i, \theta^i)\right\}_{i=N-P+1\ldots N}$ (where the data were possibly reordered) such that for data in $\mathcal{D}_1$ the likelihood $p(F(u^i)|\theta^i, \varphi)$ is close to zero for all but single $\tilde{\varphi}_1$ and likewise for $\mathcal{D}_2$, then we obtain the following upper bound for the log likelihood of observing the data:

$$\begin{aligned}
\sum_{i=1}^{N} \log p(F(u^i)|\theta^i) &= \sum_{i=1}^{N} \log \sum_{\varphi} p(F(u^i)|\theta^i, \varphi) p(\varphi|\theta^i) \\
&= \sum_{i=1}^{N-P} \log \sum_{\varphi} p(F(u^i)|\theta^i, \varphi) p(\varphi|\theta^i) + \sum_{i=N-P+1}^{N} \log \sum_{\varphi} p(F(u^i)|\theta^i, \varphi) p(\varphi|\theta^i) \\
&= \sum_{i=1}^{N-P} \log p(F(u^i)|\theta^i, \tilde{\varphi}_1) p(\tilde{\varphi}_1|\theta^i) + \sum_{i=N-P+1}^{N} \log p(F(u^i)|\theta^i, \tilde{\varphi}_2) p(\tilde{\varphi}_2|\theta^i).
\end{aligned}$$

We thus have effectively two separate maximization problems, which we can proceed to solve in either of the likelihood maximization strategies presented above. The reasoning extends to an arbitrary number of subsets. We note that i indexes *observations*, hence that $\theta^i = \theta^j$ for $i \neq j$. It could very well be that *every* θ^j is repeated. If that is the case, there are two concurrent physical processes, which we would discover by fitting each subset of the data.

Combining separate models The method produces effectively two models for the same dataset. In order to recombine these models, assume for simplicity that every observed θ is in both $\mathcal{D}_1$ and $\mathcal{D}_2$, the same number of times. Then the likelihood is

$$\begin{aligned}
\sum_{i=1}^{N} \log p(F(u^i)|\theta^i) &= \sum_{i=1}^{N/2} \log p(F(u^i)|\theta^i, \tilde{\varphi}_1) p(\tilde{\varphi}_1|\theta^i) + \sum_{i=N/2+1}^{N} \log p(F(u^i)|\theta^i, \tilde{\varphi}_2) p(\tilde{\varphi}_2|\theta^i) \\
&= \sum_{i=1}^{N/2} \log p(F(u^i)|\theta^i, \tilde{\varphi}_1) + \sum_{i=N/2+1}^{N} \log p(F(u^i)|\theta^i, \tilde{\varphi}_2) + \\
&\sum_{i=1}^{N/2} \log p(\tilde{\varphi}_1|\theta^i) + \sum_{i=N/2+1}^{N} \log p(\tilde{\varphi}_2|\theta^i).
\end{aligned}$$

If we use the second method (maximizing a lower bound of the likelihood), we use the sum of the distance to the nearest neighbors of each datum as a measure of likelihood of the data associated with each process.

[13] We chose a two dataset partition for ease of presentation but the reasoning extends straightforwardly to an arbitrary number of sets.

A remark on learning the feature space

As we noted in Section 3.4.3, learning the relationship between θ and the parameters of a differential equation is a considerable simplification. We do so by *choosing* a feature projection with the desired characteristics.

It is certainly possible to *learn* this feature projection, given enough data. To understand what is at stake, consider the setting of Section 3.4.4: we are to use the *same* number of training examples to maximize the likelihoods $\sum_{i=1}^{N} \log p(F(u^i)|\theta^i, \tilde{\varphi})$ and $\sum_{i=1}^{N} \log p(\tilde{\varphi}|\theta^i)$, where now F is to be learned.

To ground intuition, consider the problem of learning a linear mapping — the number of parameters of the linear mapping is a lower bound of the more general task. Consider the first task, that of learning the relationship between the parameters of the differential equation and the physical observables. Typically, the number of parameters of a differential equation n_φ is of order zero, and the number of physical observables n_θ is of order one. The number of parameters to learn a linear mapping between these spaces $n_\varphi \times n_\theta$ is thus of order one. As for the second task, consider that physical fields are typically discretized n, and typically take values on a m-polytope, for a total number of features of n^m. In the task of predicting laser patterns, we have $n = 224$ and $m = 2$. The number of features is of order three, which brings the number of parameters of the linear mapping between the two spaces to order seven — seven orders of magnitude more than in the previous case.

If the number of training examples is large, the two tasks can be solved in the sense that a small bound on generalization error can be found. But if the number of training examples is small — which we can define in terms of providing an acceptable bound to generalization for the first task — then the second task, since the bound on true risk is given by an increasing function of complexity, cannot be acceptably solved with the same number of data.

3.5 Predicting novel laser patterns with few data by integrating partial physical information

We now apply the framework described in the previous section to predicting new laser patterns. This problem is strictly in the scope of inverse problem theory, which aims at estimating physical model parameters based on observations, with the added difficulty of having few observations at a single moment in time and only a partial model of the physical process: the Swift-Hohenberg (SH) equation [SH77], a 4th-order partial differential equation on the plane which can be seen as a maximally symmetric model of convection.

In spite of Hypotheses 3.1 and 3.2 the problem remains severely ill-posed, and biased, since the SH prior is only an approximate model of the dynamics. Our general strategy to tackle this problem relies on finding a feature transformation F to remove some of this degeneracy.

We integrate the physical information in the SH equation in two ways: our first approach is based on training a Deep Neural Network surrogate of a SH solver [LLF98; RPK19] on great number of solutions of the SH equation, in the image of F. To our best knowledge, this is an original approach. Neural Network surrogates have been shown to provide accurate solutions of Partial Differential Equations (PDE) solvers at a fixed computational cost, and were applied successfully to notoriously difficult problems such as the three-body problem [Bre+19]. We then learn h, the mapping from laser parameters θ to SH parameters φ by backpropagating through the differentiable surrogate to minimize the mean squared error in feature space. Finally, we use the solver on the output of h in

order to produce a novel pattern, given a set of laser parameters.

Our second strategy to integrate SH physical information is to label experimental data with the SH solver parameters of its nearest neighbor, in the image of F, amongst a great number of pre-generated solutions of the SH equation. This dramatically simplifies the problem of learning the relationship h between laser parameters θ and SH parameters φ, since φ is low-dimensional. As in the first case, we use the solver on the output of h in order to, given a set of laser parameters, produce a novel pattern.

Our experiments show that the second approach yields better results than the first, with good agreement between experimental and generated images (see Figure 3.22). As we showed in Section 3.4, the second approach corresponds to maximizing a lower bound of the likelihood of the first, and the error that we incur can be controlled by the expected distance between pre-generated solutions of the SH equation in feature space.

3.5.1 The Swift Hohenberg equation: introduction and qualitative analysis

The Swift Hohenberg equation was first presented in [SH77] and [PM80] in the context of Rayleigh-Bénard convection. It is a 4th degree partial differential equation governing the time evolution of a certain real field $u(x,t)$, by defining the relationship between it spacial and time derivatives. In one dimension, with $\mathcal{N}[u]$ representing some nonlinear functional of u, we have, in adimensional form:

$$\frac{\partial u}{\partial t} = (\epsilon - 1)u - 2\frac{\partial^2 u}{\partial x^2} - \frac{\partial^4 u}{\partial x^4} + \mathcal{N}[u] \tag{3.5.1}$$

The term $\mathcal{N}[u]$ is a nonlinear term that controls the growth of the instability, the simplest choice being $\mathcal{N}[u] = u^3$. With this choice, if u is a solution of the equation, then so is $-u$; and $u = 0$ is a solution.

It can be seen seen by linear stability analysis that small perturbations of the $u = 0$ solution will be amplified selectively: specifically, when $\epsilon > 0$, Fourier modes whose wave vector norm lies on a certain interval centered at 1, the width of which depends on ϵ, will be amplified, while all others will be attenuated. The left endpoint of the interval not being zero, this is a called a type I_s instability [CH93]. This type of selective attenuation leads to the formation of patterns by selecting perturbations with certain periodicities. Since the selection of wave vectors depends only on the norm, the patterns formed via this mechanism are isotropic.

In our application, it will prove convenient to break the symmetry with respect to $u \to -u$ by introducing a term γu^2. The quadratic term allows small amplitude destabilization and the existence of the hexagonal patterns which we observe experimentally, while the negative cubic term, which dominates for large amplitudes, controls the magnitude of the instabilities, which would otherwise grow without bound. The modified form of the Swift-Hohenberg equation used in this work, also known as the Haken model [Hak77], is thus

$$\frac{\partial u}{\partial t} = (\epsilon - 1)u - 2\frac{\partial^2 u}{\partial x^2} - \frac{\partial^4 u}{\partial x^4} + u^3 - \gamma u^2 \tag{3.5.2}$$

Pattern selection: intuition To provide intuition on how a simple equation motivated on symmetry grounds originates the complex patterns observed experimentally, we examine the relationship between the time derivative and each of the terms on the right-hand side *separately*. We

shall see that it is the balance between the several terms of the SH equation which provides the complexity leading to the formation of patterns, as we illustrate in Fig. 3.8.

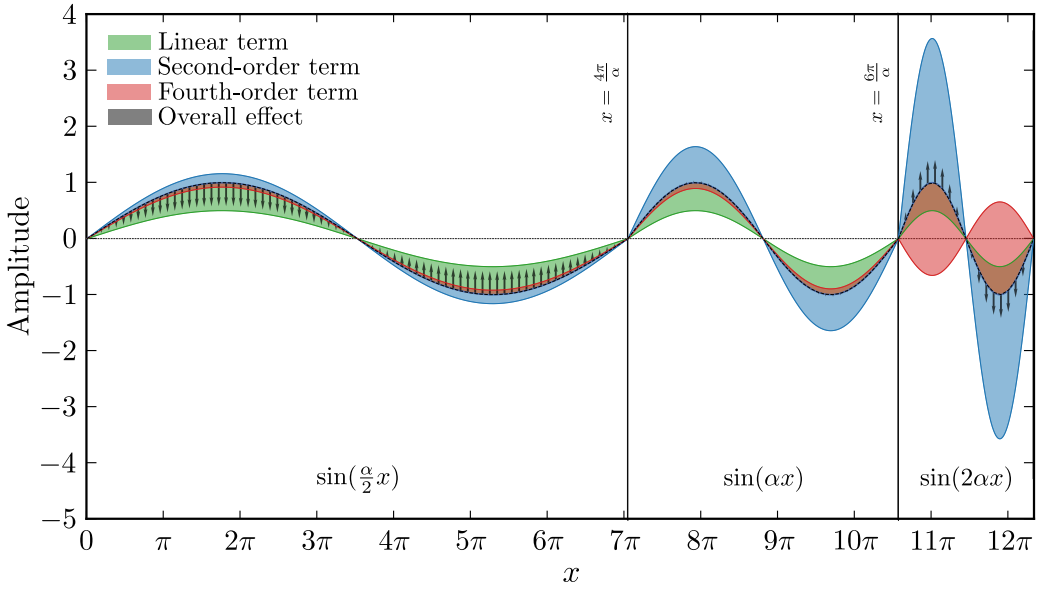


Figure 3.8: Action of the linear part of the one-dimensional Swift Hohenberg equation $\partial_t u = \epsilon u - (1+\partial_x^2)^2 u + \mathcal{N}[u]$ on three Fourier modes $u = a(t)\sin(kx)$. For small $a(t)$, ignoring the nonlinear part and to first order in time, $\partial_t u = (\epsilon - 1 + 2k^2 - k^4)a(t)\sin(kx)$. The zeros of the factor on the right hand-side are $k = \sqrt{1 \pm \sqrt{1+(\epsilon-1)}}$. For $\epsilon = 0.5$ and $k_1 = \sqrt{1-\sqrt{0.5}}$ the leftmost zero, the overall derivative of the mode $\sin(k_1 x)$ is zero, as can be seen on the center plot. Modes with lower frequency will be attenuated, as can be seen on the leftmost plot, where we show the magnitude of the various terms on $\sin(0.5k_1 x)$. Finally, the mode $\sin(2k_1 x)$, for which the frequency lies between the zeros of the factor is amplified, as can be seen on the rightmost plot.

Consider first $\frac{\partial u}{\partial t} = (\epsilon - 1)u$. Since $\epsilon \in [0, 1[$, u will increase in time where $u < 0$, and decrease where $u > 0$, in proportion to $|u|$. If the field evolution were determined by this term alone, the amplitude of u for a perturbation of the zero solution would be everywhere attenuated with time.

As for the $\frac{\partial u}{\partial t} = -2\frac{\partial^2 u}{\partial x^2}$ term, because of the negative sign, the value of u increases in regions that are convex (peaks) and decreases in regions that are concave (troughs). Determined by this term alone, perturbations of the zero solution would increase in magnitude, and more so where the frequency of u is high than where it is low.

The evolution under the fourth-order term is similar: the value of u decreases in time in regions where the fourth spacial derivative is positive and increases where it is negative. Fourth order derivatives are more difficult to visualize, so we examine its action on $\sin(q_n x)$, where q_n are integer multiples of $\frac{2\pi}{L}$, since it is a well-known fact from Fourier analysis that any sufficiently well-behaved odd f function of period L can be written as weighted sum (superposition) of these functions (the Fourier modes).

Since the derivative can be taken term by term, we examine each f_n individually. With $a_n \in \mathbb{R}$

we have

$$f(x) = \sum_n a_n \sin q_n x := \sum_n a_n f_n$$

The second derivative contribution of f_n to the Swift Hohenberg dynamics is $2q_n^2 f_n$: f_n will increase in time where $f_n > 0$ and decrease where $f_n < 0$ (as discussed above), large wavelength features changing in amplitude faster than small wavelength ones. The fourth derivative term's contribution is $-q_n^4 f_n$: f_n will decrease where $f_n > 0$ and increase where $f_n < 0$, proportionally to the inverse fourth power of its wavelength: large wavelength modes will see their amplitude change faster than small wavelengths.

To determine the overall effect, we sum up the contributions for each mode $\epsilon - 1 + 2q_n^2 - q_n^4$. The sign of this coefficient will determine if f_n gets amplified (negative) or attenuated (positive). We conclude that modes will be amplified if $1 + \sqrt{\epsilon} \leq q^2 \leq 1 + \sqrt{\epsilon}$, which determines the range of characteristic inverse wavelengths of the features in the observed patterns. We conclude by noting that this amplification is without bound, and that the nonlinear part $\mathcal{N}[u]$ plays the important role of controlling this growth.

3.5.2 The Swift-Hohenberg equation as a model of pattern formation for Rayleigh-Bénard convection

The full derivation of the Swift Hohenberg equations in [SH77] is rather involved and outside the scope of this work, but the general idea is simple, and thus we include an overview of its main lines for completeness.

The main idea behind the derivation of the Swift-Hohenberg equation in [SH77] is that of *reduced dynamics*. In the case of the [SH77] paper, this reduction is performed by adiabatic elimination. There are other techniques to do so, namely via multiple scale analysis or via normal forms derivation (see [Sid11] for a detailed derivation). The interested reader may also consult [Man83], where a similar derivation to that in [SH77] is performed in physical space rather than working in Fourier space.

In the vicinity of an instability point (a set of conditions at which a system undergoes a transition from a stable to an unstable state), small perturbations to the system no longer decay but rather grow over time. Swift and Hohenberg start in the context of Rayleigh-Bénard convection in the Boussinesq approximation. There, it is known that this instability is given in terms of a critical value of ΔT, the temperature difference between the two horizontal plate, beyond which sustained convection appears. This is often defined in terms of a critical Rayleigh number R_c, where, in the notation in [SH77], the Rayleigh number is given by

$$R = \frac{g\alpha l^3 \Delta T}{\nu\kappa}$$

here g is the acceleration due to gravity α is the thermal expansion coefficient of the fluid l is the distance between the plates at ΔT temperature difference, ν is the kinematic viscosity of the fluid, and κ its thermal diffusivity.

After rewriting the temperature equation in terms of a variable θ which describes the departure of the temperature T from the uniform gradient $\Delta t/l$, the authors introduce what is known as a Langevin forces in the conservation of energy and conservation of momentum equations in the Boussinesq approximation. These Langevin forces model random "kicks" in energy and momentum

that particles experience due to their thermal environment, which will be reflected in kicks in temperature and velocity as well.

The idea is to follow the effect of these fluctuations in the vicinity of the critical temperature. To do so, the authors linearize and then Fourier transform the Boussinesq equations for the temperature and the z-component of the velocity with the Langevin forces. This allows to identify two eigenvalues of the linearized system: one with a real part that tends to zero at the critical point (stable), and one that remains finite (unstable). The real part of its eigenvector being much greater, the stable eigenvectors thus evolve much faster than the unstable eigenvectors in the vicinity of the instability.

The key idea now is that the faster evolution of the stable eigenvectors will allow them to adapt to the evolution of the unstable nodes. In the terminology in [Hak77], the stable modes are *slaved* to the unstable modes and can thus be adiabatically[14] eliminated from the dynamics. Thus, in the vicinity of the instability point the dynamics may be *reduced* to the slow modes dynamics.

To actually perform this reduction, the authors in [SH77] proceed to represent the full non-linear dynamics in the basis of the eigenvectors of the linear system (in Fourier space). One of these equations, that of the slow mode, in the vicinity of the instability, can be seen as stationary. Solving this equation for the critical slow mode (see Appendix B in [SH77]), and plugging it back into the nonlinear equations, the authors arrive at an expression that, up to a reparameterization and up to a negligible noise term (at least in the limit of large Prandtl number ν/κ), the Swift Hohenberg equation in the eigenvector of the slow mode.

3.5.3 The Swift-Hohenberg equation as maximally symmetric model of pattern formation

Although the SH equation was introduced as a model of Rayleigh-Bénard convection as described above, it can actually be motivated on general grounds, by appealing to symmetries. Specifically, given appropriate boundary conditions, the SH equation is the simplest equation with a type I_s instability that is isotropic, translation invariant, and with invariance with respect to the $u \to -u$ substitution (see [CH93] for details). Crucially, the SH equation can be derived *from these assumptions alone* [CH93], as we shall see in the following, in an argument that is loosely based on [CH93] and [VHV94], and that we present for completion.

We begin by assuming that the field $u(\mathbf{x}, t)$ that describes a two-dimensional pattern at position $\mathbf{x}$ and time t, approximately respects the following conditions:

1. Translational Invariance: Invariance under translation $\mathbf{x} \to \mathbf{x} + \mathbf{a}$, where $\mathbf{a}$ is an arbitrary vector. This is of course an idealization and implies that u has no spacial structure and extends infinitely in all directions.

2. Isotropy: Invariance under rotation of $\mathbf{x}$.

3. Reflection Symmetry: $u \to -u$.

These symmetries are, in our experimental setting, approximately verified: (i) translational invariance is clearly approximately verified at the scale of the patterns that we consider (the center of the laser spot), as can be seen by inspection. This is explains the efficacy of the convolutional neural network extracted features, as we shall see in Sec. 3.4.4. (ii) since cross-polarization was introduced

[14] The term "adiabatic" here, which originates in thermodynamics, refers to a process that occurs so slowly that the system has enough time to adapt, thereby allowing the elimination of fast modes in favor of slow-changing ones.

specifically to make the patterns isotropic [Abo+20], we also expect this symmetry to hold within the limits of experimental reality. (iii) reflection symmetry is not expected to hold, and we shall explicitly break this symmetry introducing a quadratic term in the SH equation.

Instabilities of dynamical systems We shall consider idealized pattern-forming systems, which evolve from a uniform, translational invariant state following an equation

$$\partial_t u = \mathcal{N}[u]$$

where ∂_{x_i} means partial differentiation with respect to variable x_i, and the right-hand side is a non-linear differential operator acting upon field u. We shall be interested in how a perturbation of base state $u_p = u - u_b$ will grow over time. Since both u and $u_b = 0$ are solutions of the equation we have

$$\begin{aligned} \partial_t u = \mathcal{N}[u] \Rightarrow \\ \partial_t(u_p + u_b) = \mathcal{N}[u_p + u_b] \Rightarrow \\ \partial_t u_p = \mathcal{N}[u_p + u_b] - \partial u_b \Rightarrow \\ \partial_t u_p = \mathcal{N}[u_p + u_b] - \mathcal{N}[u_b] \end{aligned}$$

hence

$$\partial_t u_p = \mathcal{N}[u_p + u_b] - \mathcal{N}[u_b] \tag{3.5.3}$$

If the field u_p is sufficiently small, we can approximate $\mathcal{N}[u_p + u_b]$ by linearizing about u_b, which will result in keeping only terms that involve a single factor of u_p or its spatial derivatives on the right hand-side of eq 3.5.3, resulting in a linear evolution equation

$$\partial_t u_p = \left(\sum_i a_i(x,t) \partial_{k_i}^{n_i} \right) u_p(x,t) := \mathcal{L}[u_p]$$

We can look for Fourier mode solutions of this equation of the form

$$u_p(x,t) = A e^{\sigma t} e^{\alpha \cdot \mathbf{x}}$$

where $\sigma, \alpha := (\alpha_1, \cdots, \alpha_n), A$ are possibly complex. This is convenient because j-degree differentiation with respect to any spatial variable x_i becomes multiplication:

$$\partial_i^j u_p = \alpha_i^j u_p$$

and replacing it in the linearized equation for the perturbations leads to a simple polynomial equation:

$$\sigma = \sum_i a_i(x,t) \alpha_{k_i}^{n_i}$$

Consistently with the infinite-sized domain necessary for translational invariance, we assume periodic boundary conditions, that is, the spacial part of the perturbation is invariant under translations

by a certain vector $\mathbf{L} := (L_1, \ldots)$. That is, for every x_k we have $e^{\alpha_k x_k} = e^{\alpha_k (x_k + L_k)}$. This implies that $\alpha_k L_k = (2\pi i)m$ for some integer m, which implies that $\alpha_k = iq_k$ is purely imaginary. Specifically, the numbers q_k are quantized and can only take up the following values

$$q_k = m\frac{2\pi}{L_k}, \quad m \in \mathbb{Z}.$$

The vector $\mathbf{q} := (q_1, \ldots)$ is called the *wave vector*. This means that we are interested in solutions of the form

$$u_p(x,t) = Ae^{i\sigma t}e^{i\mathbf{q}\cdot\mathbf{x}}.$$

where we have redefined $\mathrm{Re}(\sigma) := \sigma$ Since the equation is linear, a general solution can be found as a superposition of particular solutions

$$u_p(x,t) = \sum_{\mathbf{q}} c_{\mathbf{q}} e^{i\sigma_{\mathbf{q}} t} e^{i\mathbf{q}\cdot\mathbf{x}},$$

where the sum goes over the set of quantized tuples of q_k found above. For the base state to be linearly stable, that is $u_b = 0$ for $t \to \infty$, all growth rates $\sigma_{\mathbf{q}}$ must be negative.

Heuristic derivation of the SH equation Since the system is rotationally invariant, the growth rate of a perturbation of the base state $\sigma(\mathbf{q})$ can only depend on q, the norm of the wave vector $\mathbf{q}$, not its orientation. Let q_c be the norm of the wave number such that, for $q > q_c$, the growth rate becomes positive and the modes become unstable, as discussed in Sec. 3.5.1. Then we can expand σ_q in the neighborhood of this critical wave number; keeping only the first even-order terms

$$\begin{aligned}\sigma(q) &\approx \sigma(q_c) + a(q - q_c)^2 \\ &= \sigma(q_c) + aq^2 + 2aqq_c + aq_c^2\end{aligned}$$

We shall be interested in equations of this type that have a stationary type-I-s instability, that is, for which the previous equation has a local maximum at $q = q_c$ for $\sigma(q_c)$, called the *control parameter*, equal to zero. This means that the constant $a < 0$.

Now since, as seen above, for Fourier mode solutions $u_{\mathbf{q}}$, we have $\sigma(q)u_{\mathbf{q}} = \partial_t u_{\mathbf{q}}$ and similarly for the spatial operators e.g. $q^2 u_{\mathbf{q}} = (q_x^2 + q_y^2)u_{\mathbf{q}} = -\nabla^2 u_{\mathbf{q}}$, then by multiplying the growth rate expansion by $u_{\mathbf{q}}$ on both sides we obtain the following equation

$$\partial_t u_{\mathbf{q}} = \left(\sigma(\mathbf{q}_c) + a\left(\sqrt{-\nabla^2} - q_c\right)^2\right) u_{\mathbf{q}}$$

The fractional derivative term $\sqrt{-\nabla^2}$ is impractical to work with outside the Fourier domain. For that reason, noting that for $\sigma(q_c)$ small, $q + q_c \approx 2q_c$, and one can approximate the right hand side of the expansion for $\sigma(q)$ as

$$\begin{aligned}\sigma(q) &\approx \sigma(q_c) + a(q - q_c)^2 \\ &\approx \sigma(q_c) + a\left(\frac{q + q_c}{2q_c}\right)^2 (q - q_c)^2 \\ &= \sigma(q_c) + \frac{a}{4q_c^2}(q^2 - q_c^2)^2\end{aligned}$$

Using the same anszats as before we obtain,

$$\partial_t u_{\mathbf{q}} = \left(\sigma(q_c) + \frac{a}{4q_c^2}\left(-\nabla^2 - q_c^2\right)^2\right)^2 u_{\mathbf{q}},$$

which can be rescaled to cast in adimensional form, in what is commonly known as the adimensional Swift-Hohenberg equation:

$$\partial_t u = ru - \left(\nabla^2 + 1\right)^2 u.$$

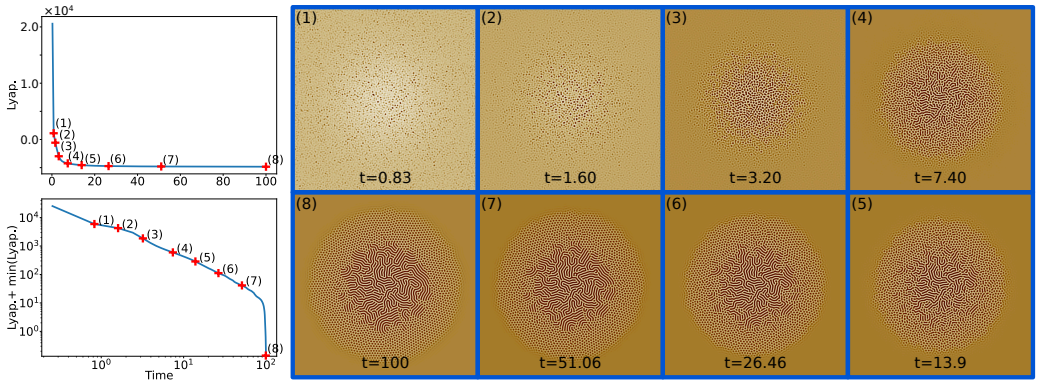


Figure 3.9: **(right)** SH solution with ϵ a centered 2D Gaussian ramp with a maximum of 0.8 and a standard deviation of 1/4 the domain size, to mimic the laser fluence distribution, and $\gamma = -1.0$, shown at several solver times t, with 1024^2 colocation points, represented as a 1024^2 heatmap (normalized to 1 since the SH eq. is adimensional); **(left, top)** SH Lyapunov functional $E[u] = \int_\Omega \frac{u}{2}\left(\nabla^4 u + 2\nabla^2 u + u\right) + \frac{1}{4}u^4 - \frac{\gamma}{3}u^3 - \frac{\epsilon}{2}u^2 d\mathbf{x}$ (a functional of the field which has the property to decrease during the dynamics) with points corresponding to the solutions on the right highlighted. Note that, as expected, E converges asymptotically to a stable value; as can be observed from the field heatmaps on the right, as it does so, organized patterns form. **(left, bottom)** SH Lyapunov functional shifted up in log-log scale; SH solver terminates when the Lyapunov functional stops decreasing.

3.5.4 The Swift-Hohenberg equation has potential dynamics

The Swift Hohenberg equation for field $u(\mathbf{x}, t)$ on a domain Ω with the periodic boundary conditions that are used in this work has potential dynamics, meaning that there is a functional E of the field u, called the Lyapunov functional, such that

$$\dot{u} = -\frac{\delta E}{\delta u} \tag{3.5.4}$$

where $\frac{\delta E}{\delta u}$ is the variational derivative of E with respect to variations δu. The functional E has the property of decreasing during the dynamics [CH93]. Since $E[u]$ is also bounded below, it converges

asymptotically to a stable value. The Lyapunov functional for the SH equation used throughout this work is

$$E[u] = \int_\Omega \frac{u}{2}\left(\nabla^4 u + 2\nabla^2 u + u\right) + \frac{1}{4}u^4 - \frac{\gamma}{3}u^3 - \frac{\epsilon}{2}u^2 d\mathbf{x} \tag{3.5.5}$$

$$:= \int_\Omega \mathcal{E}\left(u, \nabla^2 u, \nabla^4 u\right) d\mathbf{x} \tag{3.5.6}$$

as can be verified straightforwardly by calculating the functional derivative $\frac{\delta E}{\delta u}$ using the Euler-Lagrange equations

$$\begin{aligned}\frac{\delta E}{\delta u} &= \frac{\partial \mathcal{E}}{\partial u} + \sum_{i=2,4} (-1)^i \nabla^i \cdot \frac{\partial \mathcal{E}}{\partial\left(\nabla^i u\right)} \\ &:= \underbrace{\frac{\partial \mathcal{E}}{\partial u}}_{\text{A}} + \underbrace{\nabla^2 \cdot \frac{\partial \mathcal{E}}{\partial\left(\nabla^2 u\right)}}_{\text{B}} + \underbrace{\nabla^4 \cdot \frac{\partial \mathcal{E}}{\partial\left(\nabla^4 u\right)}}_{\text{C}}\end{aligned}$$

And since $\nabla^2 \cdot \frac{\partial \mathcal{E}}{\partial(\nabla^2 u)} = u$ and $\nabla^4 \cdot \frac{\partial \mathcal{E}}{\partial(\nabla^4 u)} = u/2$, we get

$$\begin{aligned}A &= \frac{\nabla^4 u}{2} + \nabla^2 u + u + u^3 - \gamma u^2 - \epsilon u \\ B &= \nabla^2 u \\ C &= \frac{\nabla^4 u}{2},\end{aligned}$$

which implies, replacing in eq. (3.5.4)

$$\begin{aligned}\dot{u} &= -\nabla^4 u - 2\nabla^2 u - (1-\epsilon)u - u^3 + \gamma u^2 \\ &= \epsilon u - (\nabla^2 + 1)^2 u - u^3 + \gamma u^2,\end{aligned}$$

which is indeed the generalized SH equation.

The fact that E decreases during the dynamics (cf. Figure 3.10) will allow us to check for divergence by calculating it at fixed iteration intervals, as we explain in the following section.

3.5.5 A pseudo-spectral second order solver for the SH equation

In this section, we describe the finite-difference solver that we use in this thesis to integrate the knowledge of the Swift-Hohenberg equation into the machine learning model. We shall also use this solver indirectly in the form of a surrogate neural network, which motivates some of the implementation choices that we describe below. We choose a second-order pseudo-spectral method, providing a good compromise between accuracy and speed. A pseudo-spectral method is a split-step method, a technique which we explain briefly below following mainly [Lev07] and [Yos90]. We use one spacial dimension in our discussion for ease of presentation, as it generalizes straightforwardly to the plane. We assume periodic boundary conditions throughout.

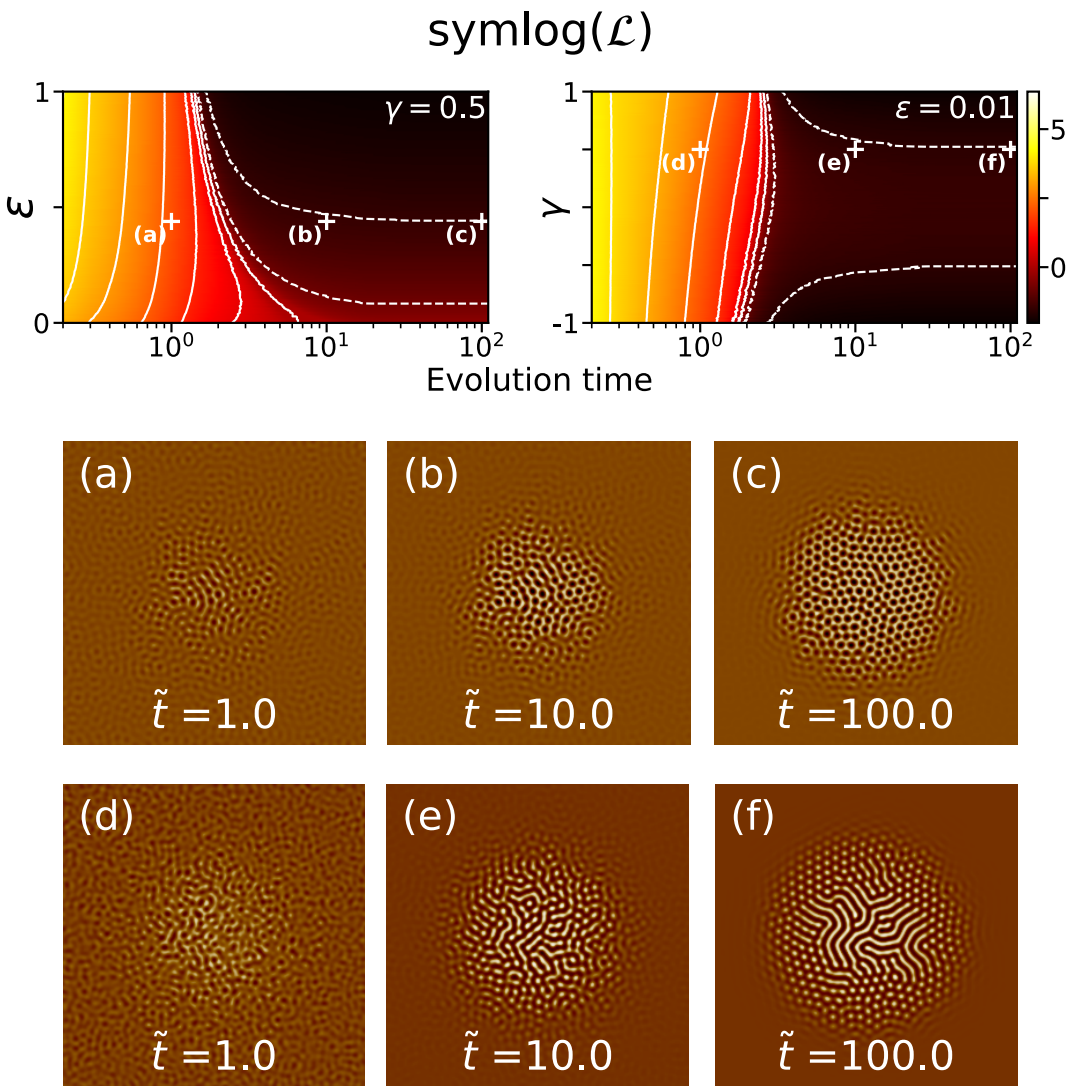


Figure 3.10: Lyapunov functional of the generated field solutions of the SH equation as a function of evolution time $\tilde{t}$ for fixed ϵ and γ (ϵ a centered 2D Gaussian ramp to mimic the experimental laser fluence distribution consistent with Sec. 4.5.1), depicted as a heatmap in symlog scale (symlog(x) is the odd function that equals $\ln x$ if $x > 1$ and x if $0 \leq x \leq 1$), for independent initial conditions. Lyapunov functional evolution is largely independent of initial conditions and decreases during dynamics. The SH equation is able to reproduce, among others, highly symmetric hexagonal solutions (top), as well as labyrinthine solutions surrounded by nanopeaks.

Operator splitting

The SH equation can be written in terms of the sum of the action of a linear $\mathcal{L}$ and a non-linear $\mathcal{N}$ differential operators acting on u

$$\dot{u} = \mathcal{L}[u] + \mathcal{N}[u] \tag{3.5.7}$$

with the action of $\mathcal{L}$ given by $\mathcal{L}[u] = (\epsilon - (1 + \partial_x^2)^2)u$ and the action of the nonlinear part defined as $\mathcal{N}[u] = \gamma u^2 - u^3$.

The idea is to discretize and integrate each of these parts in turn, which explains the name of the technique. Doing so allow us to solve the non-linear part in Fourier space, where its action reduces to multiplication, a considerable reduction in computational cost. The nonlinear part can be solved using a straightforward explicit method which is easy to implement. The solution $u(t_{n+1})$ of the equation above at a later time $t_{n+1} = t_n + dt$ can be obtained from that at $u(t_n)$ via the exponential of the operator $\mathcal{L} + \mathcal{N}$ [Lev07]

$$u(t_{n+1}) = \exp\left(dt(\mathcal{L} + \mathcal{N})\right)u(t_n).$$

Integrating each part separately amounts to using the approximation

$$\exp\left(dt(\mathcal{L} + \mathcal{N})\right) \approx \exp\left(dt\mathcal{L}\right)\exp\left(dt\mathcal{N}\right). \tag{3.5.8}$$

The error that we incur in doing so can be established using the *Baker-Campbell-Hausdorff formula* (see [Hal03] for a proof and [Yos90] for applications to higher-order integrators), which states that for any non-commutative operators X and Y, the product of the two exponentials can be expressed in terms of the exponential of a single operator Z

$$\exp(X)\exp(Y) = \exp(Z)$$

where Z is given in terms of the commutators of X and Y in all except the linear term

$$Z = X + Y + \frac{1}{2}[X, Y] + \frac{1}{12}([X, [X, Y]] + [Y, [Y, X]]) + \frac{1}{24}[X, [Y, [Y, X]]] \cdots \tag{3.5.9}$$

Applying this formula with $X = dt\mathcal{L}$ and $Y = dt\mathcal{N}$ we conclude, since the commutator $[dt\mathcal{L}, dt\mathcal{N}] = dt^2[\mathcal{L}, \mathcal{N}] \neq 0$, that the approximation eq3.5.8 is order one — independently of the order of the methods that we choose for each of the individual parts. Specifically, we have

$$\begin{aligned} u(t_{n+1}) &= \exp\left(dt\mathcal{L} + dt\mathcal{N}\right)u(t_n) \\ &= \exp\left(dt\mathcal{L}\right)\exp\left(dt\mathcal{N}\right)u(t_n) + \mathcal{O}(dt). \end{aligned}$$

Second-order pseudo-spectral solver

It is possible to increase the order of approximation at the expense of extra intermediate steps [Yos90]. In this work, we use an order two splitting scheme, known as *Strang splitting* [Str68], which consists in taking a half step with the linear operator, a full step with the nonlinear operator, and a final half step with the linear operator[15]:

$$u(t_{n+1}) = \exp\left(\frac{dt}{2}\mathcal{L}\right)\exp\left(dt\mathcal{N}\right)\exp\left(\frac{dt}{2}\mathcal{L}\right)u(t_n) + \mathcal{O}(dt^2).$$

[15] We examined a symmetric 3-fold Strang composition method [Yos90] of order four, but the improvement was not sufficient for our purposes to justify the increase in execution time, which stems not only from the extra time stepping, but from the higher-order methods for each of the individual steps.

Strang splitting increases the order of ordinary time-splitting to two provided we choose order two methods to discretize each of the individual steps. We thus integrate the nonlinear part using Ralston's method, an order two explicit Runge-Kutta method with the Butcher tableau (see e.g. [Lev07] for notation)

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 2/3 & 2/3 & 0 \\ \hline & 1/4 & 3/4 \end{array}$$

and integrate the linear part in Fourier space, where spacial derivatives amount to multiplication, using the trapezoidal rule, with Butcher tableau

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array},$$

the result then being transformed back to the original space. Using the Fast Fourier Transform [CT65] to calculate the discrete Fourier transform allows us to take this otherwise costly step at quasi-linear time complexity.

With $T(\mathcal{L}, u^t, k)$ and $R(\mathcal{N}, u^t, k)$ representing the Trapezoidal and Ralston's numerical methods solving, respectively, $\dot{u} = \mathcal{L}[u]$ and $\dot{u} = \mathcal{N}[u]$ over a time step of k starting with data u^t, and denoting the discrete Fourier transform of a field u by v, our pseudospectral scheme reads

$$\begin{aligned} u^* &= \mathcal{F}^{-1}\left\{T(\mathcal{L}, v^t, \frac{dt}{2})\right\} \\ u^{**} &= R(\mathcal{N}, u^*, dt) \\ u^{t+1} &= \mathcal{F}^{-1}\left\{T(\mathcal{L}, v^{**}, \frac{dt}{2})\right\} \end{aligned}$$

where u^*, u^{**} are intermediate solutions.

Stability and adaptive time-stepping

The Trapezoidal rule is an order two method with a region of absolute stability including the entire left half of the complex plane (which makes it suitable for the solution of stiff equations [But16; Lev07]); stability of the split step method is thus determined by that of the explicit step.

We use an adaptive time step of the order of inverse of the spectral norm of the Jacobian of the nonlinear stepping operator to control the stability of the explicit scheme, and improve the time performance of our solver. We control for divergence by examining the Lyapunov functional at fixed iteration intervals. As explained above, a growth of the Lyapunov functional implies divergence. When this happens, we go back to the time before Lyapunov functional growth, divide the time step by two, and proceed with the time stepping. This method is simple but allows us to automate the solver with minimum input on our part, which is important given the number of images that we need to generate.

Training a differentiable surrogate of the SH solver

For our method it is convenient to train a differentiable surrogate of the SH solver described above *in feature space*. Although strictly unnecessary, as we could alternatively use automatic differentiation, introducing a surrogate allows dramatic decrease in training time [RPK19] for learning the

relationship between the SH parameters and the laser parameters. Faster forward problem solutions provided by the surrogate will also prove convenient in evaluating feature choices by cross validation.

The surrogate consists in a neural network $f_\omega : \varphi \to F$ which learns the mapping between 4-dimensional φ (SH equation parameters, evolution time, and scale) to the projection of the SH solver generated solutions in feature space, by minimizing the MSE. We use a 5 hidden-layer neural network with GeLU activations [HG16], which we initialize using He initialization [He+15] and regularize using weight decay. The number of units of the hidden layers are $2^4, 2^6, 2^8, 2^{10}, 2^{12}$. The architecture was chosen by cross validation on a fraction of the data. The neural network was trained for 1000 epochs with early stopping (cf. Fig. 3.11).

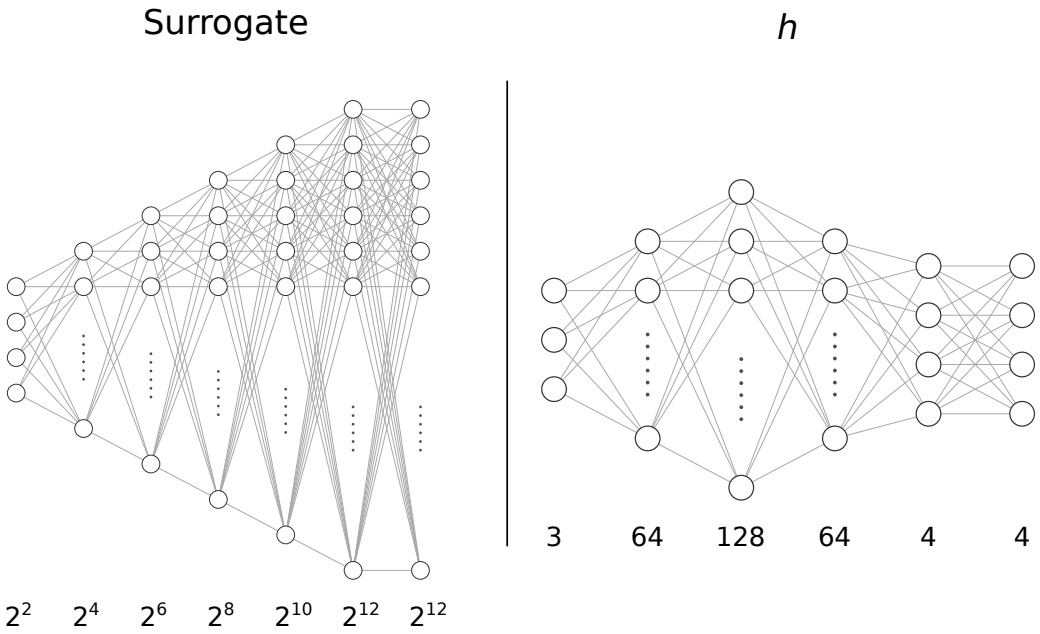


Figure 3.11: Neural network architectures: **(left)** differentiable SH surrogate, mapping from $\varphi \in \mathbb{R}^4$ (Swift Hohenberg equation parameters, evolution time, and scale) to feature space ($\mathbb{R}^{224\times224}$). **(right)** mapping h from laser parameters $\theta \in \mathbb{R}^3$ (laser Fluence, delay between pulses, and number of pulses) to $\varphi \in \mathbb{R}^4$.

3.5.6 Choosing a feature space to learn patterns

We would like our framework presented in Section 3.4, which crucially depends on the existence of a feature space in which the dependency of initial conditions is weak, to be generally applicable in the early stages of the experimental process, where there is neither enough physical knowledge to handcraft this invariance explicitly nor enough data to learn it directly. These two constraints motivate the use of features extracted by a large pre-trained model on a diverse dataset.

Off-the-shelf feature space projector

We choose, as feature projector, the first before last layer of a pre-trained VGG16 [SZ15], a deep convolutional network (CNN) model trained on ImageNet [Den+09], a dataset of over 15 million labeled high-resolution images belonging to roughly 22,000 categories. See figure 3.12 for details. Any complex model trained on a complex dataset like ImageNet is likely to acquire biases that depend on the dataset itself. Some of these may actually be good [Gei+18], but since they were not specifically controlled for, they could be undesirable for our task. In this sense, VGG16 appears to be a reasonable candidate in the context of our application.

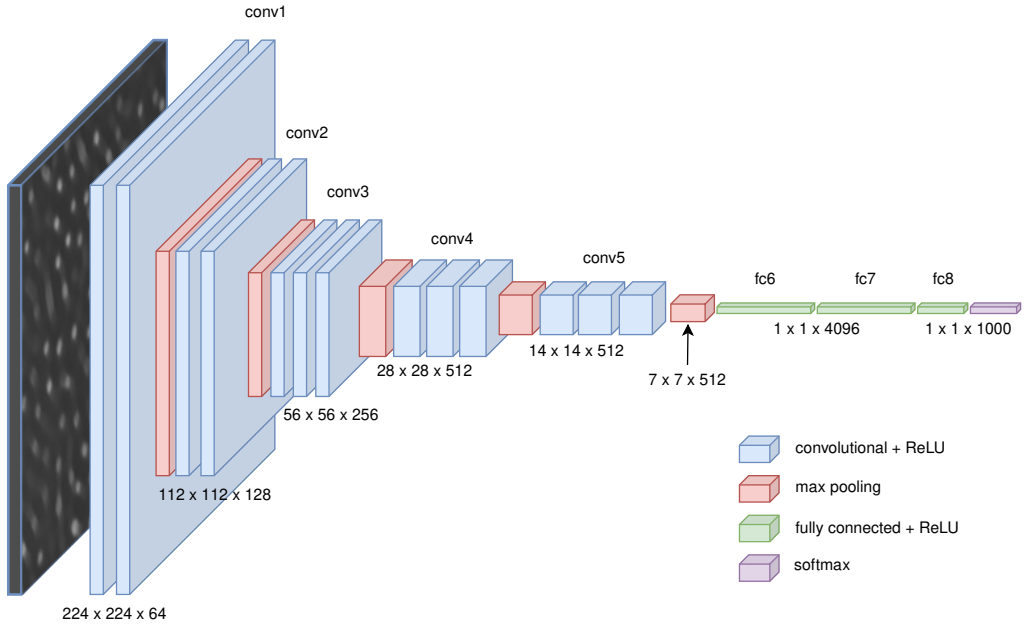


Figure 3.12: VGG16 network architecture. Our baseline features are extracted using layers "conv1" to "fc6" (inclusive).

Deep Convolutional Neural Networks achieve state-of-the-art performance on image classification tasks [Sze+16; He+16]. VGG16 in particular achieves 92.7% top-5 test accuracy on ImageNet, which is the main motivation for our choice, since the representation that a deep convolutional model needs to build (edges, textures, colors and combinations thereof), in order to do well in such a classification task should be complex enough to represent the Swift-Hohenberg equation patterns as well. We further justify our choice by noting that this same feature extractor was used successfully to localize casting defects in (grayscale) X-ray images [Fer+17], and classify weld defects, both of which bare some similarities to our task.

Learning scale with the features of a scale-invariant network

There is still the problem of pattern scale, that needs to be learned. The success of convolutional neural networks and the scale-invariant nature of common image classification tasks led to extensive

work in precisely the opposite direction: learning scale *invariant* features, rather than learning scale.

Convolutional neural networks such as VGG16 are not truly scale invariant, though. Scale invariance is rather learned implicitly, even without a specific scale-invariant design [LV15], by training on datasets such as ImageNet, in which instances of the same class are represented at different scales. To obtain truly multi-scale scale invariant representations, we need to resort to local descriptors such as SIFT, which are popular in image processing [Sze10], or recent deep learning techniques which focus specifically on scale invariance and equivariance [KSJ14; Mar+18; WW19; GG19], or impose it as a particular group symmetry [CW16].

Although learned implicitly, learned scale invariance for large convolutional models trained on ImageNet such as ResNet50 [He+16] and InceptionV3 [Sze+16] is quite good, as it was found recently that the probability of the correct class is approximately invariant to input size [Gra+21]. The authors show, however, that scale invariance varies with depth, information about scale being chiefly present at intermediate layers, with invariance reached just before the softmax layer, and early layers focusing on local textures and small object parts. The authors then go on to show that by pruning the layers where the scale invariance is learned there are gains on a medical imaging task which, like our regression task, depends on scale.

Bearing this in mind, we prune the last two layers of a pre-trained VGG16, resulting in features encoding scale information and with enough complexity to represent the patterns of the Swift-Hohenberg equation. The resulting feature space has 4096 dimensions in the base case.

Building the datasets

We have two data types: one consisting of real SEM images, the other of SH-generated images.

Because experimental manipulation is costly and time-consuming, the first dataset is small. It consists of 78 SEM images labeled with the laser parameter values F_p peak *laser Fluence* (in J/cm^2), *time delay* Δt (in $10^{-15}s$) and *number of pulses* N, of an area roughly 5 µm^2 size with a resolution of 237 pixels per µm.

The second dataset consists of a set of real fields generated by the SH equation with periodic boundary conditions on a square of side 224 according to the following procedure: we first sample uniformly the order parameter $\epsilon \in (0, 1)$, the symmetry breaking parameter $\gamma \in [-2, 2]$, and the system size $l \in (8, 25)$ (in units of 2π); we seed the square with pointwise uniform initial conditions in $[0, \sqrt{\epsilon}]$ and evolve it according to the SH solver 3.5.5 until convergence, as assessed by the time derivative of the Lyapunov functional; we keep a maximum of ten snapshots of this evolution at regular solver time intervals t. Each of the resulting field is labeled with the tuple ϵ, γ, l, t.

Pre-processing Since we shall ultimately compare real and generated images, we need to make a choice regarding image normalization. The experimental images are obtained via SEM microscopy, their intensity being preset. As for the generated images, the initial perturbation has a maximum amplitude of $\sqrt{\epsilon}$, and after a typical evolution time of $\frac{1}{\epsilon}$ [CH93], in the linear approximation, the maximum amplitude will remain a function of ϵ. To see this, note that the maximum growth rate $\sigma(q^2) = \epsilon - 1 + 2q^2 + q^4$ is at $q^2 = 1$, which corresponds to a maximum amplification at evolution time $1/\epsilon$ of $e^{\epsilon/\epsilon} = e$, regardless of our chosen ϵ. Multiplying an ϵ-dependent maximum initial amplitude by a constant factor remains ϵ-dependent. The actual significance of such an amplitude depends also on the size of the domain, which is one of the parameters of the Swift-Hohenberg generated fields. Furthermore, since we take ten images at arbitrary time snapshots, the relationship between the maximum amplitude across fields is complex. This renders any normalization to our data other

than image-wise difficult to establish (or essentially meaningless). This arbitrariness of levels is actually compatible with VGG16, in the sense that we expect that the ImageNet photographs that it was trained to classify were acquired with arbitrary levels and exposure.

We therefore normalize each image individually and subsequently transform the resulting array to pixel intensity space. Since VGG16 takes color images as its input, we use copies of our grayscale images as the input for the remaining two channels, before the final pre-processing of the input provided as a Keras [Cho+15] method (centering and normalizing each color channel with respect to ImageNet, and flipping RGB to BGR).

Subsampling VGG16 takes 224x224 pixel images as inputs. In order to keep as much information as possible, we sub-sampled a maximum of ten such images at random orientations for each SEM image instead of downsampling the data. We allow for some variation in the number of samples due to varying quality of SEM images, some of which have large patches consisting mostly of noise, which were removed.

Splitting the dataset As described in Section 3.4.4 one of the key assumptions in our framework is that the likelihood is peaky, possibly in different subsets of the data. We do observe that several SEM images have patches of two superimposed patterns of different length scales, an assertion that can be confirmed by analyzing their Fourier power spectrum, as shown in Figure 3.13. Since the

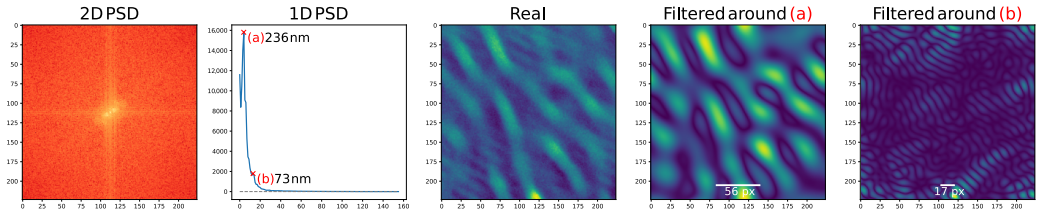


Figure 3.13: Pattern superposition: SEM image Fourier spectrum has two modal frequencies corresponding to two different patterns of different characteristic wavelengths. **(2D PSD)** is a heatmap of the logarithm of the Fourier power spectral density of the **(Real)** image. **(1D PSD)** is the fourth power (to exaggerate peaks) of the 1D PSD, obtained by azimuthally averaging 2D PSD along radii from the origin [Lu11]. Peaks marked in red are automatically extracted using a SimPy [Meu+17] method and correspond to the physical wavelengths displayed in the labels. The two images on the right are obtained from the real image by filtering out all wavelengths except the ones corresponding to the 1D PSD peaks, by multiplying it in Fourier space with a "Gaussian annulus" centered at the center of the image in Fourier space with a diameter equal to the wavelength of the peak. Image **(Filtered around (a))** highlights the "top" pattern, whereas image **(Filtered around (b))** highlights the "bottom" pattern.

SH equation is a single scale model, we built two expert-constructed datasets, "bottom" and "top", with respectively 435 and 550 samples taken at random orientations, with a pattern that was found, in superimposed patterns, either bottom or on top. We concatenated the data into an "full" dataset consisting of 985 images. A visualization of the parameter ranges of each dataset can be seen in Figure 3.14.

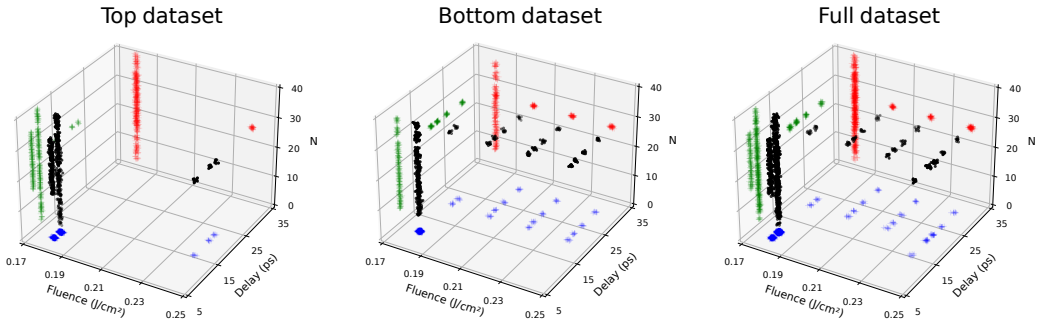


Figure 3.14: Real data laser parameters 3D scatter plots, with jitter for ease of visualization, for the "top", "bottom", and "full" datasets with, respectively 550, 435, and 985 data points. Data is represented in black, with projections on the Fluence/Delay plane in blue, Fluence/N plane in red, and Delay/N plane in green.

Selecting the best feature space

In order to select the best feature space, we performed a KNN clustering task on the experimental images using Euclidean distance on each instance of the extracted features, and asked the domain experts to assess the quality of the extracted clusters, as described in 3.4.4.

We evaluate feature transformations consisting in variations of the features extracted by pruning the last two layers of a pre-trained VGG16: a version denoted "normalized" in which zero-variance features were removed and the remaining features scaled to unit variance; a version denoted "scale variant", where we explicitly introduce scale information by concatenating the VGG16 extracted features with features with scale information consisting of a 1D power summary of the 2D power spectrum of each image obtained by azimuthally averaging the 2D power spectrum (PSD) along radii from the origin [Lu11]; and finally, a version denoted "aligned", in which a simple feature space alignment method [Fer+13] was used to align the distributions of the generated and real images.

Since normalization and alignment, as well as experts and MSE evaluations, are dataset-dependent, we compare each of these feature mappings in each dataset: "bottom", "top", "full". We present a selection of the most relevant variations below.

Expert clustering results As can be observed in Table 3.1 feature evaluation tends to favor the VGG16-extracted features without further transformation. We show cluster assignment for a random sample of images in the full dataset for the case of the VGG16-extracted features in Figure 3.15. Adding the PSD features generally reduces the expert-assessed clustering quality, which is consistent with the findings in [Gra+21]. If we choose to add the PSD features, however, we observe that the clustering quality improves by performing feature subspace alignment between real and SH-generated images, which can be explained by the fact that the PSD of real images contains small-scale information that is not present in SH-generated images, and that VGG16-extracted features are invariant to small-scale "noise".

Dataset	PSD	Normalized	Aligned	Accuracy
full	False	False	False	**0.778**
full	False	False	True	0.776
full	True	True	True	0.699
full	True	True	False	0.671
full	True	False	True	0.634
bottom	False	False	False	**0.953**
bottom	False	False	True	0.952
bottom	True	True	True	0.812
bottom	True	True	False	0.712
bottom	True	False	True	0.793
top	False	False	False	0.857
top	False	False	True	**0.864**
top	True	True	True	0.855
top	True	True	False	0.770
top	True	False	True	0.709

Table 3.1: Expert-assessed cluster quality for different feature choices. Best result for each dataset in bold. Adding the PSD features does not improve the expert-assessed clustering quality. If we choose to add the PSD features, however, the clustering quality improves by performing feature subspace alignment between real and SH-generated images.

3.5.7 Learning $h : \theta \to \varphi$

Having access to a SH solver, the key task is learning the function h from laser parameter space to SH parameter space.

We begin by noting that the equation that we used throughout this chapter is adimensional, and that it can be derived on symmetry grounds only (cf. Sec. 3.5.3). For that reason, the *scale* is a hyperparameter of our solver, which also needs to be learned. The experimental setting described in [Nak+21a], implies that there is a pattern solidification time that is independent of that of the process (which is chiefly controlled by the order parameter). This makes it unlikely that the observed patterns will be the long-time solutions of the SH equation: we are most likely observing transients. This is also apparent from inspection (e.g. top real image, third column in Figure 3.1. We thus choose as SH parameters the SH equation parameters ϵ and γ, the SH solver domain size l given in units of 2π, and the solver evolution time t. The laser parameters are F_p the laser peak fluence in units of J/cm^2, the time delay between pulses Δt in picoseconds, and the number of laser pulses N.

We use the two methods described in Section 3.4.3 to learn h: directly 3.4.3 and by maximizing a lower bound 3.4.3.

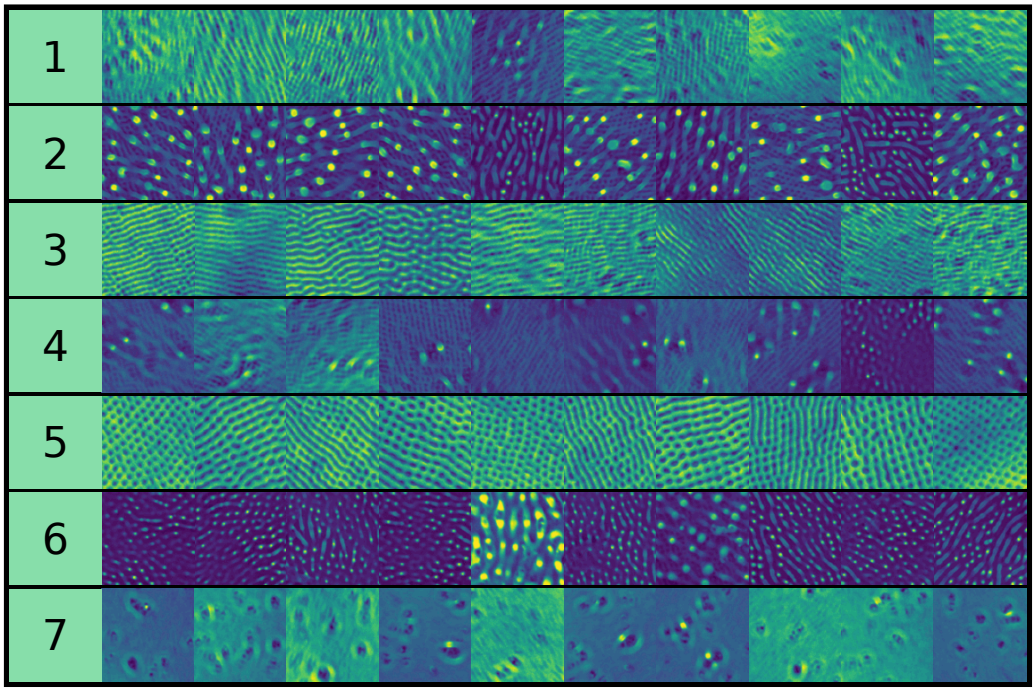


Figure 3.15: 10 random samples for each of the 7 clusters obtained by k-means clustering (k=7) for the f000 features, which consist of the full dataset projected into the VGG16 feature space without further modification.

Learning h directly

In order to learn h directly, we use the pre-trained surrogate $\text{Solver}_{\text{surr}} : \varphi \to F$ described in Section 3.5.5 with frozen weights, with the following objective:

$$\min_{\alpha} \frac{1}{M} \sum_{i=1}^{M} \left\| \text{Solver}_{\text{surr}} \circ h_{\alpha}(\theta^i) - F(I^i) \right\|_2^2 \tag{3.5.10}$$

where h_{α} is a neural network parameterized by α, and M is the number of experimental data. We use a 4-hidden layers neural network with GeLU activations, which we regularize using weight decay. The number of units of the hidden layers are 64, 128, 64, and 4 (cf. Fig. 3.11). The network was trained over 1000 epochs with early stopping. We illustrate this procedure in Figure 3.16.

Learning h indirectly

In order to learn h indirectly, we first "train" a 1NN model with respect to the Euclidean distance on the pre-generated SH solutions I^i with parameters in the set Φ_{pregen} (as described in Section 3.5.6) projected to feature space, and assign to each experimental datum the SH parameters of its nearest

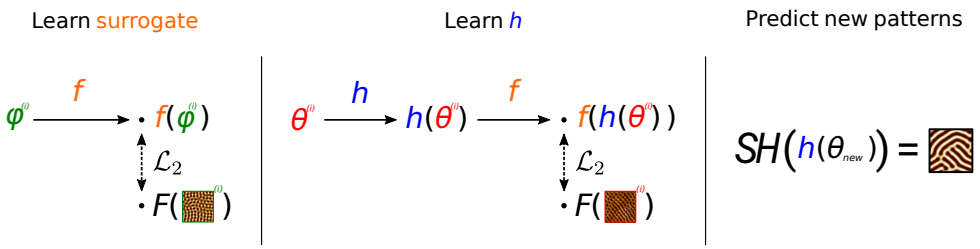


Figure 3.16: Learning h directly: **(left)** we begin training a surrogate f (orange) of the SH solver $\mathrm{SH}(\cdot)$ in the image space of F, on a great number of pre-generated $\varphi^{(i)}, F(\mathrm{SH}(\varphi^{(i)}))$ pairs (green), where $\varphi^{(i)}$ are SH parameters and $F(\mathrm{SH}(\varphi^{(i)}))$ is the image in feature space of the solution generated by our solver with parameters $\varphi^{(i)}$, by minimizing the mean squared error in feature space; **(center)** we then learn $h : \theta \to \varphi$ (blue) with real data $\left\{\theta^{(i)}, I^{(i)}\right\}_{i=1\dots M}$ (red) by minimizing the mean squared error in feature space; **(right)** finally, we generate patterns for unseen θ_{new} using the learned h and the solver: $\mathrm{SH}(h(\theta_{\text{new}}))$

.

neighbor. Explicitly:

$$\varphi^i = \underset{\varphi \in \Phi_{\text{pregen}}}{\arg\min} \left\| \text{Solver}(\varphi) - F(I^i) \right\|_2 \tag{3.5.11}$$

We then learn h by minimizing the mean squared error in feature space

$$\min_{\alpha} \frac{1}{M} \sum_{i=1}^{M} \left\| \varphi^i - h_\alpha(\theta^i) \right\|_2^2 \tag{3.5.12}$$

where h_α is a support vector regressor with RBF kernel, $C = 1.0$ and $\epsilon = 0.1$, and M the number of experimental data. We illustrate this procedure in Figure 3.17.

3.6 Experimental results

In this section we present the experimental results. We begin by showing a comparison of cross-validation scores of the two methods in Section 3.5.7 as well as the feature mappings and datasets defined in Section 3.5.6. Cross validation results in feature space show high variance, which is possibly a consequence of few training data, but could also be due to problems in training the surrogate, which is used to evaluate the scores (see the direct model discussion on Section 3.6.2 for details). In addition, the variety of feature transformations renders direct comparison between scores in different spaces difficult to interpret. As explained in Section 3.5.6, in our setting, the choice of feature space is best done by expert-evaluated cluster accuracy.

We then show a selection of predictions of the learned models and a map of parameter space for each dataset. We shall see that the choice of method is faced with much the same difficulties as the choice of feature mapping. We observe that better cross validation scores do not necessarily lead to better predictions, and that this is a consequence of the severe constraints of our problem.

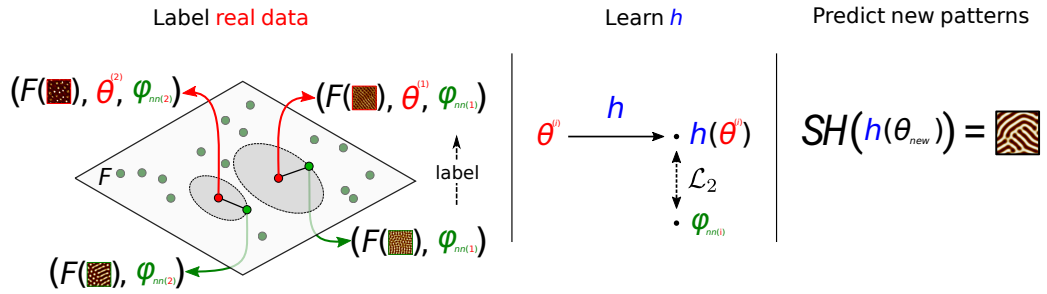


Figure 3.17: Learning h indirectly: **(left)** we begin by labelling each experimental datum i (red) with the SH parameter $\varphi_{\mathrm{nn}(i)}$ of its nearest neighbor, in the image space of F, among a great number of pre-generated solutions of the SH equation (green) ; **(center)** we then learn $h : \theta \to \varphi$ with data $\left\{\theta^{(i)}, \varphi_{\mathrm{nn}(i)}\right\}_{i=1\ldots M}$ by minimizing the mean squared error in parameter space; **(right)** finally, we generate patterns for unseen θ_{new} using the learned h and the solver: $\mathrm{SH}(h(\theta_{\mathrm{new}}))$

.

We also observe that our best model is able to recover the main features (shape and scale) of test data, and that the learned model is simpler for data where there is no pattern superposition. We show evidence that splitting the dataset improves the quality of the model, which suggests concurrent multiscale SH processes taking place, as explained in Section 3.4.4.

3.6.1 Cross validation

Ideally, in order to compare the different feature mappings and methods, we would like to perform 10-fold cross validation of mean squared error in feature space, which is closest to the task. The problem with this approach is that the image space of different feature mappings has different dimensions, which makes mean squared error comparison for different feature mappings meaningless[16]. In order to circumvent this difficulty, we can compare MSE in the 4-dimensional image of h, which we call *SH parameter space*. Although cross-validation scores become comparable across different feature spaces, this strategy is not without problems. Indeed, MSE in feature space is not necessarily a good measure of feature quality and the comparison across *methods* is unfair, since the indirect method relies on optimizing MSE in parameter space to with respect to the nearest neighbors, which is advantageous.

Baseline method, parameter space We define a baseline method as the "regressor" which predicts, for each datum, the SH parameters of its nearest neighbor in feature space. Instead of presenting the 10-fold cross validation MSE results for this method, we actually present the leave-one-out cross validation results, the latter being a lower bound of the former. The reason is expedience, since the latter is simply the average squared distance in between data in each dataset.

[16] An alternative would be to normalize mean squared error by the variance, but this runs into the same problems that we discussed in Section 3.4.4 and yields inconsistent results. If we use this method to normalize the data in Table 3.3, the best feature space for the "full" dataset would be f111 which underperforms the expert-chosen f000 features by 7%.

Cross validation in parameter space As we can observe in Table 3.2, cross-validation results for the direct method are compatible with the expert-based clustering results in 3.4.4, although the strength of this conclusion is limited by the high variance in the scores.

We also observe that cross-validation scores for the indirect method are best for features for which the PSD was appended and real and generated data features are aligned. Although the mean is strictly lower than in the case where the features are further normalized, the score ranges intersect.

A possible explanation for the better performance of the aligned feature choices is that the method crucially relies on the quality of the nearest neighbors. Fitting nearest neighbors in high dimensions is difficult, since all points tend to be equidistant. Since aligned feature spaces are lower dimensional, their performance for this method should be better.

	baseline	direct	1NN
b000	8.019	**2.065±0.629**	0.975±0.202
b001	8.019	2.979±1.146	0.870±0.205
b101	8.019	3.701±1.565	**0.745±0.283**
b111	8.019	3.180±2.739	0.765±0.335
t000	8.015	**2.169±0.976**	0.989±0.150
t001	8.015	5.207±5.226	0.914±0.203
t101	8.015	5.323±2.589	**0.609±0.142**
t111	8.015	4.513±6.354	0.615±0.122
f000	8.008	2.351±0.852	0.993±0.950
f001	8.008	2.814±1.814	0.912±0.901
f101	8.008	3.693±3.240	**0.659±0.201**
f111	8.008	**2.006±0.330**	0.694±0.167

Table 3.2: 10-fold cross validation mean and standard deviation of MSE in SH parameter space for different methods (baseline, direct, 1NN), datasets, and feature spaces. Each row is labeled as: *dataset*, *Includes PSD*, *Normalized*, *Aligned*; 1 denotes True and 0 False. The b101 row for example, lists the MSE and standard deviation in SH parameter space for the *bottom* datataset (b) where features include the PSD (1), are not normalized (0) and are aligned to with respect to the bottom dataset (1). Best scores for each method for each dataset are highlighted in bold. Note that SH parameter space is the same for every row and column.

Baseline method, feature space We define a baseline method as follows: the "regressor" which predicts the features of the nearest neighbor in feature space. As in the case of the parameter space baseline above, we report the leave-one-out cross validation MSE score, which is a lower bound of the 10-fold cross-validation score, consisting of the mean squared distance in between data in each dataset.

Cross validation in feature space We present the 10-fold cross validation scores in feature space for several choices of modified VGG16 features, datasets, and methods in Table 3.3. We note the high variance of the scores as well as the difference in scores across feature spaces spanning twelve orders of magnitude. We also note that the "direct" method is always better, as we evaluated it using the SH solver surrogate (doing 10-fold cross validation using the solver directly is impractical

and would increase the variance even more), which makes for an unfair comparison with the 1NN "indirect" method.

	baseline	direct	1NN
b000	4.55e04	**9.92e00±7.23e-01**	1.46e01±5.27e-01
b001	3.70e04	**8.93e01±9.73e00**	1.37e02±7.7e00
b101	2.75e09	**3.95e06±1.44e06**	7.99e09±1.45e09
b111	2.17e-03	**1.08e-04±2.00e-04**	7.98e-03±1.21e-03
t000	4.54e04	**1.08e01±9.52e-01**	1.43e01±8.04e-01
t001	3.71e04	**7.22e01±6.05e00**	9.36e01±5.08e00
t101	6.25e09	**3.92e07±1.02e08**	9.13e09±1.88e09
t111	5.51e-03	**1.07e-05±1.46e-06**	1.25e-02±1.39e-03
f000	4.73e04	**1.08e01±1.13e00**	1.43e01±2.73e-01
f001	3.70e04	**9.57e01±1.04e01**	1.44e02±4.79e00
f101	6.93e09	**9.09e06±1.17e07**	9.10e09±1.10e09
f111	6.00e-03	**1.38e-05±4.48e-06**	1.02e-02±1.05e-03

Table 3.3: 10-fold cross validation mean and standard deviation of MSE in feature space for different methods (baseline, direct, 1NN), and feature spaces, for the full dataset. Each row is labelled as: *dataset*, *Includes PSD*, *Normalized*, *Aligned*; 1 denotes True and 0 False. The f000 row for example, lists the MSE and standard deviation in SH parameter space for the *full* datataset (f) where features do not include the PSD (0), are not normalized (0) and are not aligned to with respect to the bottom dataset (0). Best scores across rows are highlighted in bold. Note that comparing across columns is meaningless, as MSE are calculated in different feature spaces.

3.6.2 Model predictions

In this section we present the model-predicted SH parameters for the expert-selected features for unseen data from the "top", "bottom" and "full" datasets, using the indirect method. We also present the direct method model-predicted SH parameters (for the full dataset only) which are inconsistent, and provide a possible explanation for this behavior.

Indirect method

Comparing the model predictions for the "top", "bottom", and "full" datasets (cf. Fig. 3.19, 3.20 and 3.21 respectively), we note that the first two are much simpler than the predictions learned on the combined dataset, which is consistent with the discussion about pattern superposition in Section 3.5.6 and the hypothesis that there is more than one SH process at play.

For all datasets, we note that the models struggle to extrapolate to regions for which there is no experimental data, in particular for laser fluence: predictions were generated for $F_p \in (0, 0.5)$ for models trained on fluence data in $(0.18, 0.24)$; but as can be seen in the Figures above, we only show the fluence interval $(0.1, 0.32)$ because outside this range predictions are essentially constant. As for the time delay, experimental data $\Delta t \in [8, 25]$ but we manage to observe interesting prediction in the $[0, 50]$ range. For the laser parameter N (number of pulses), in spite of most experimental data being sampled at $N = 25$, models manage to extrapolate to unseen N based on two sets of experimental series at $\{(F_p = 0.18 \text{ J/cm}^2, \Delta t = 10 \text{ ps}, N)\}_{N=6\ldots36}$ and $\{(F_p = 0.18 \text{ J/cm}^2, \Delta t = 8 \text{ ps}, N)\}_{N=15\ldots33}$

(the vertical lines in Figure 3.14). This opens up the possibility to improving predictions with sparse *additional* experimental data, with a focus on the F_p laser parameter.

The SH parameter that shows the largest variation in predictions across values of N is the evolution time t parameter — which is what we would expect, since a larger number of pulses implies a process that is more extended in time. On the other hand, for all datasets, the complexity of the learned relationship increases with the number of observations, with sharp boundaries of rapidly varying parameters appearing where there is enough data.

Also noteworthy is the correlation between the various SH parameters l, γ, ϵ, t, as can be seen in Figure 3.18, which implies that one cannot design laser patterns freely. This correlation changes as

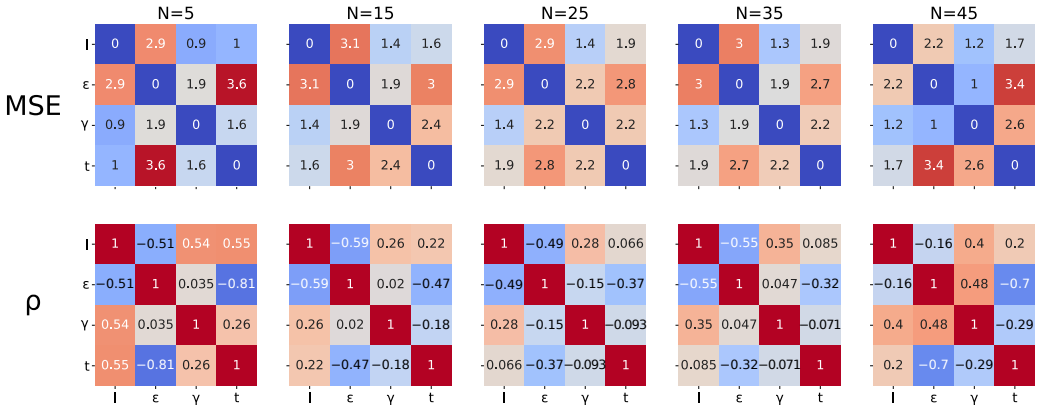


Figure 3.18: Mean squared error (top row) and Pearson's coefficient (bottom row) between SH parameters l, ϵ, γ, t for indirect method predictions for the f000 features, for several values of N (columns).

the number of pulses N of the laser varies, which again testifies to the complexity of the underlying process.

System size l The parameter l is inversely correlated with the pattern characteristic size. Consistently with observations, l increases with N. Interestingly, this increase is not uniform: it is greater for large l regions in laser parameter space for low N, than for small l regions. The rapidly transitions between l regions that can be observed for all values N are of possible interest to applications, in particular where other parameters are constant, as it would allow one to control the characteristic size of the particular pattern defined by the other SH parameters.

Symmetry breaking parameter γ The SH parameter γ determines whether we observe holes or bumps. Of particular interest is the fact that for the "top" dataset, the transition from holes to bumps is in the Δt direction, whereas for the "bottom" dataset, the change is in the F_p direction. This suggest that either two fundamentally different SH processes or a non-SH process are at play.

Order parameter ϵ To a larger parameter ϵ, farther from onset, there correspond less ordered patterns, since there is a greater number of non-attenuated Fourier modes. The large ϵ low order patches at high fluence/low delay for the "bottom" dataset are consistent with this fact and match

observations. For the "top" dataset, however, we observe an ordered pattern region of low ϵ and small l at low values of delay, which is more challenging to interpret. For the "full" dataset, the complexity of the ϵ isosurfaces in the central region of and around the isosurface of $\gamma = 0$ is consistent with observations where a lower symmetry pattern is superimposed on a highly symmetric grid pattern (cf. Fig. 3.1).

Evolution time t For constant l, ϵ, γ, symmetry increases with evolution time t as highly symmetric patterns require a large t to produce from a uniformly random state.

On the other hand, for all datasets, evolution time t tends to increase with N, which is consistent with the physical situation, as a large number of pulses increases the time the physical system is in a driven state. This increase, however, is not uniform across fluence, delay pairs: indeed we observe that the area of the region in laser parameter space of relatively long evolution time decreases with N.

Generating novel patterns Instead of relying on the SH parameter interpretations above, one can use the SH solver to generate new patterns, as illustrated in Figure 3.17. We present the generated solutions from unseen laser parameters together with the corresponding real SEM image and nearest neighbors in feature space in Figure 3.22. Comparing solutions generated by the model from unseen test data to real SEM images allows further interpretation. The model does quite well for the "Stripes" and "Hexagons" group, as there is a single pattern to learn. For the two other groups there is more than one pattern/feature superimposed on the SEM image. Note that the model does predict one of the observed patterns with the correct scale and shape but, by design, it cannot possibly predict the other. For the "Stripes and Bumps" group, for instance, we observe nanopeaks and lower-frequency stripes among the nearest neighbors, whereas the model predicts the lower frequency pattern only. For "HSFL and Humps" group, where the difference in pattern frequency is more pronounced, again the model focuses on the lower frequency pattern; the high-frequency pattern is no longer among the nearest neighbors.

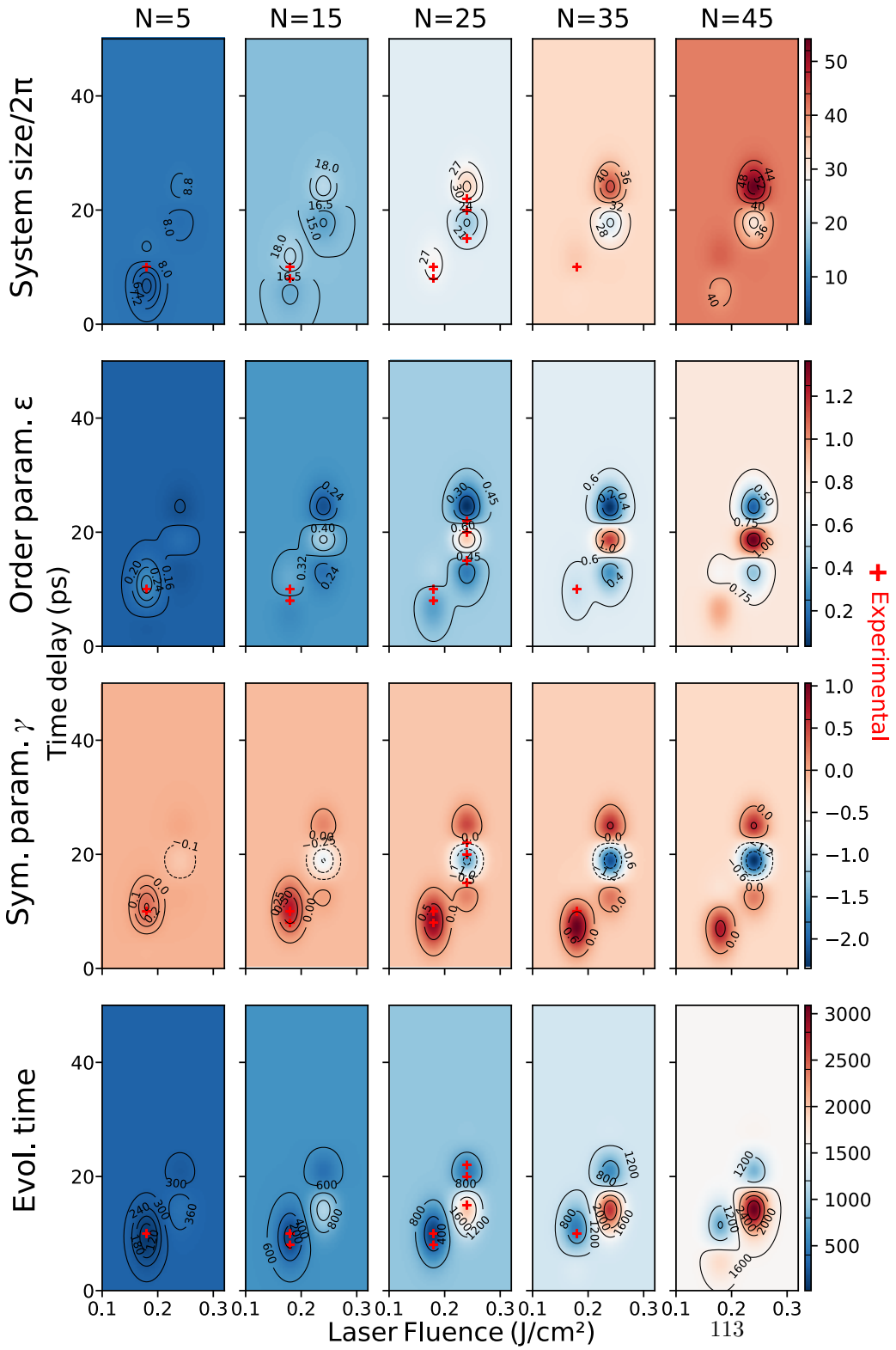


Figure 3.19: Each plot shows the predictions of the indirect model, trained on the "top" dataset, of a single SH parameter, as a heatmap (top to bottom: system size in multiples of 2π; order parameter ϵ; symmetry breaking parameter γ; solver evolution time) as a function of laser fluence, time delay, and number of pulses (respectively, x-axis and y-axis, and column). Experimental points are overlaid on each plot.

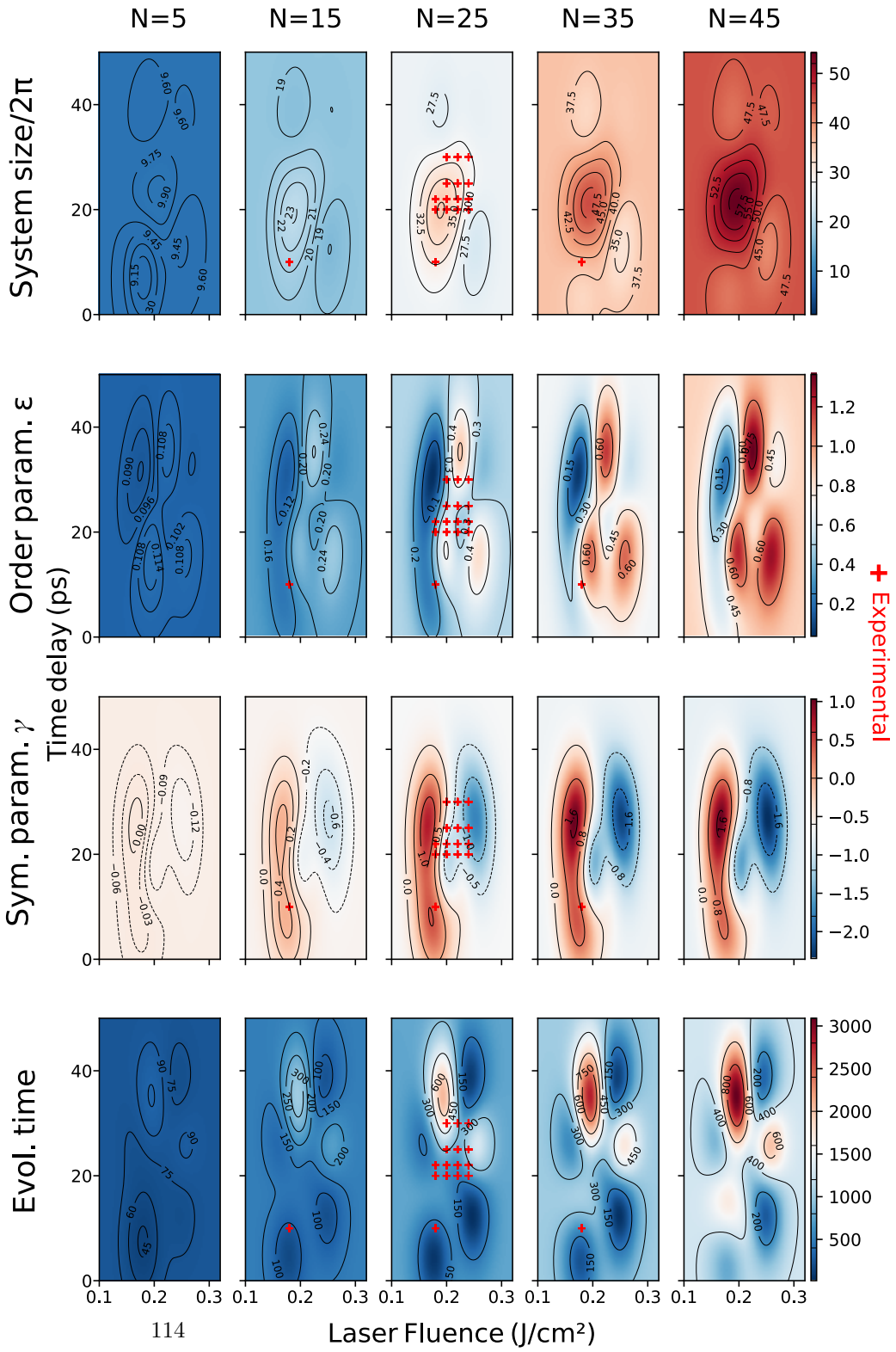


Figure 3.20: Each plot shows the predictions of the indirect model, trained on the "bottom" dataset, of a single SH parameter, as a heatmap (top to bottom: system size in multiples of 2π; order parameter ϵ; symmetry breaking parameter γ; solver evolution time) as a function of laser fluence, time delay, and number of pulses (respectively, x-axis and y-axis, and column). Experimental points are overlaid on each plot.

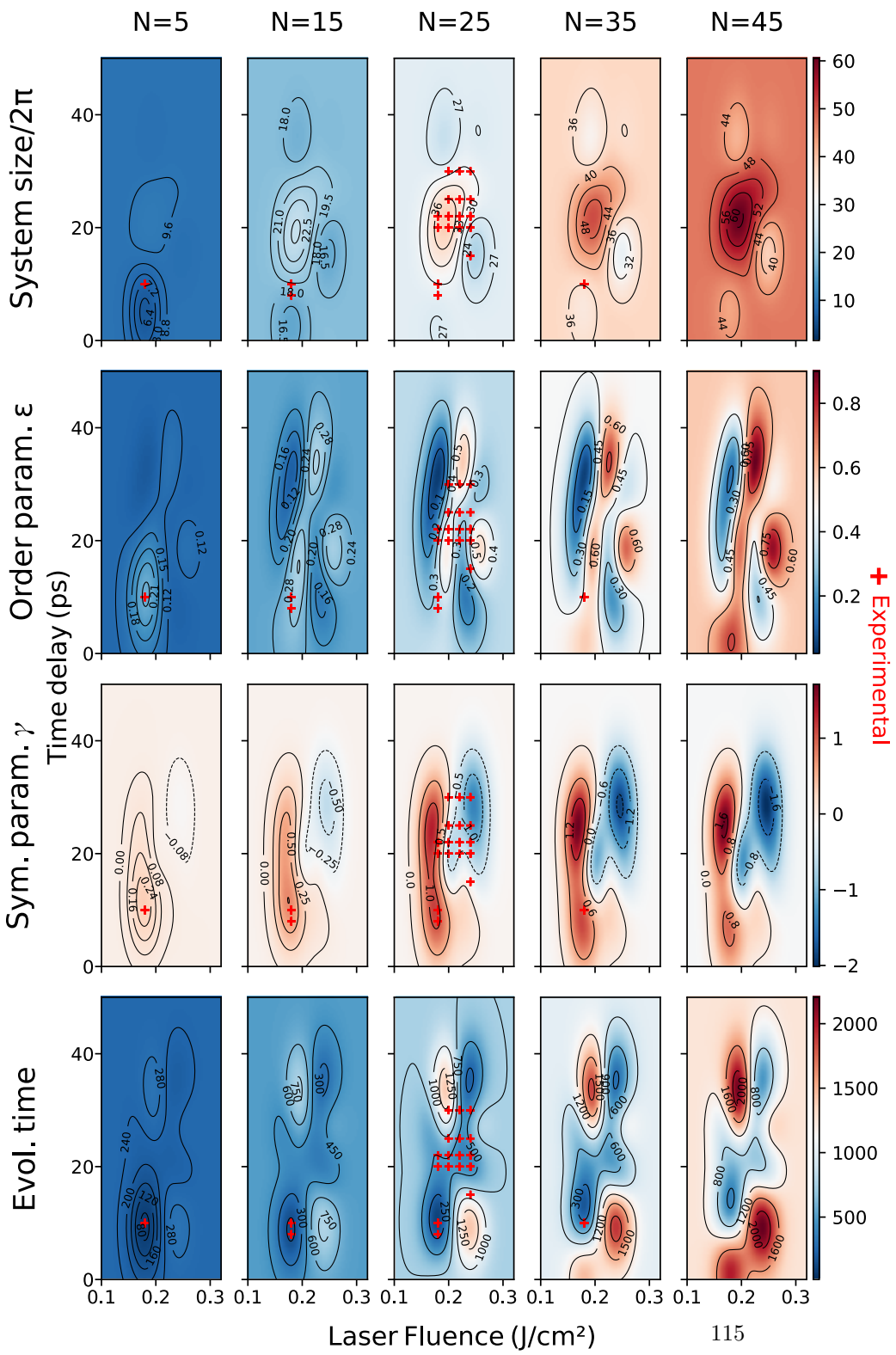


Figure 3.21: Each plot shows the predictions of the indirect model, trained on the full dataset, of a single SH parameter, as a heatmap (top to bottom: system size in multiples of 2π; order parameter ϵ; symmetry breaking parameter γ; solver evolution time) as a function of laser fluence, time delay, and number of pulses (respectively, x-axis and y-axis, and column). Experimental points are overlaid on each plot.

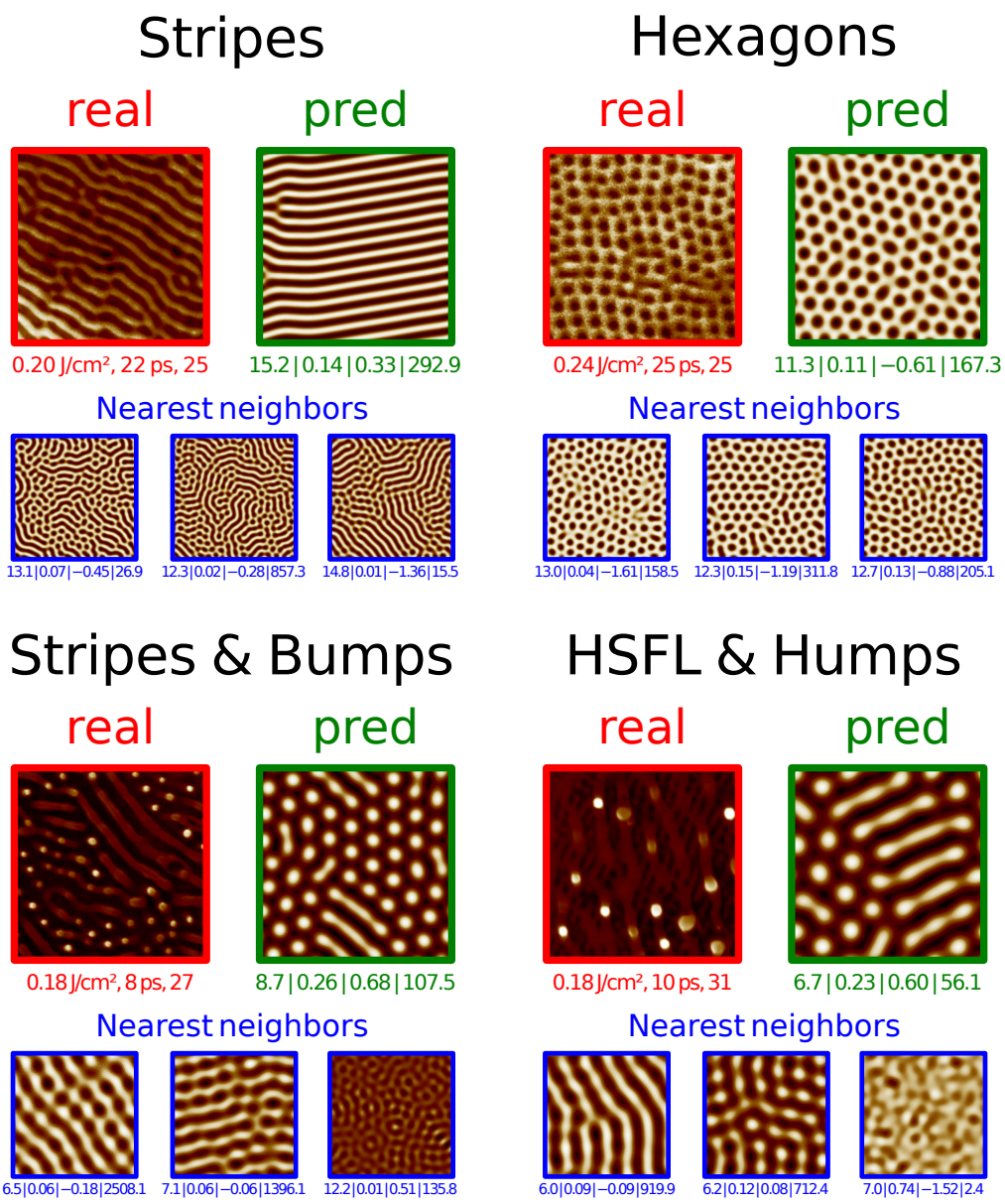


Figure 3.22: Each group shows experimental SEM images (**red**, never seen by the model), model predicted images (**green**, trained on the "full" dataset using the indirect method), given the same laser parameters and three nearest neighbors of the former among solver generated images (**blue**). Image labels, left to right: $F_p, \Delta t, N$ (real images); l, ϵ, γ, $\tilde{t}$ (other images). All images are 224 by 224 pixels; for real images, $1\mu m \approx 237$ pixels. Model's predictions are better than the nearest neighbor, since they integrate global information. On the "Stripes and Bumps" group, we observe nearest neighbors with different length scales, suggesting concurrent multi-scale SH processes taking place. Only one of these can be recovered by the ML model (the stripes, rather than the nanobumps), which integrates single-scale SH knowledge. On the "HSFL and Humps" group, the real image has a top, low frequency pattern and a finer grid pattern underneath. The model predicts the former, which is also the only the among the nearest neighbors.

Direct method

While the surrogate trains well on SH-generated data, as can be seen in Figure 3.23, the h model, which trains on few real data, is unstable. Although this small dataset artifact could partially

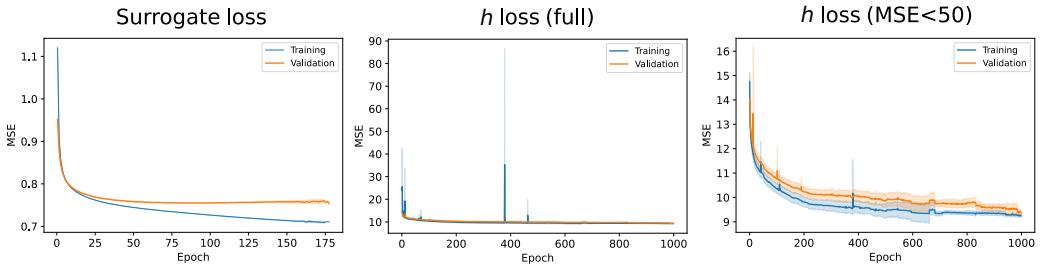


Figure 3.23: Direct method training and test (90/10) loss curves (mean and 95% confidence interval):**(left)** surrogate training on the large dataset of SH-generated data is stable, with narrow confidence interval bands; in all runs, early stopping was activated considerably earlier than the 1000 epoch limit **(center)** the $h : \theta \to \varphi$ model, trained on few real data (the "full" dataset), on the other hand, is unstable; this behavior could be attributed to the dataset size, but the problem remains even for typical runs **(right)** detail of $h : \theta \to \varphi$ training, with outliers for which training MSE exceeds 50 removed. A number of runs for which no early stopping was "activated" is evident, which further evidences training instability.

explain why the direct method fails to yield accurate predictions, the problem remains even for "aligned" datasets and for typical training runs, closer to the center of the confidence interval, as can be seen in Figure 3.24. It is known that surrogates using fully connected architectures often fail to achieve stable training by gradient descent and produce accurate predictions [Rai18; Zhu+19; FT20], and it is believed that this phenomenon is due to the difficulty of deep fully-connected networks in learning high-frequency functions [WYP22]. We note that since we learn the surrogate on feature space, we cannot assess generated solutions accuracy directly. That being said, this unstable behavior is consistent with the very large variance observed in the cross validation experiments in feature space 3.6.1.

While methods have been proposed to improve this behavior, either by introducing an adaptive learning rate [WTP21] or by adjusting the weight of each term in the loss according to the singular values of the Neural Tangent Kernel [WYP22], we do not use them in this work, as the indirect method provides satisfactory results, is robust, and straightforward to implement.

3.7 Conclusion

In this work we integrated partial physical knowledge in the form of a PDE, the SH equation, to solve the problem of predicting novel nanoscale patterns in femtosecond irradiated surfaces. We showed that in the case of a self-organization process, the dual inverse problem of estimating state and equation parameters simplifies by choosing a feature transformation in the image space of which the initial conditions play a less important role. In the case where data is few and not time-series and the physical knowledge is only partial, this transformation can neither be learned nor derived:

we use as transformation the higher-order features of a CNN pre-trained on a large dataset for a broad task. We proposed a principled approach to choosing such a feature transformation, and an expert based quality measure of the features as well.

We integrated the PDE knowledge by implementing a fast and accurate second-order pseudospectral solver of the SH equation and then by using a great number of pre-generated solutions to learn a surrogate in feature space, on the one hand, and to label the few experimental data with SH parameters of the nearest neighbors in feature space, in the other. This technique allowed us to learn the relationship between laser parameters and SH parameters (with which novel laser patterns can be generated via the solver), a relationship that can be used as an experimental tool to guide new pattern discovery.

This led us to make a number of observations. First, in spite of the good agreement between the partial SH model predictions and experimental data, we also found evidence that there is more than one SH process at play. This leaves the door open to either using a generalized SH model that integrates several length scales, or to exploring possibilities to combine multimodal single-scale data into a superimposed solution. Second, we observed that pattern features are not independent; finding novel patterns requires searching laser parameter space "creatively" by looking at regions where some SH parameter varies and some other does not. Third, we noted that although the model does not extrapolate well, it is still able to learn interesting features from few data. This opens the door to a dialectic approach to novel pattern discovery: one could simply acquire new experimental data iteratively in order to fill-in the gaps in the laser parameter space until the predictions stabilize; since the SH model is already trained and data is pre-generated, integrating new experimental data only requires retraining a simple model on few data.

The main challenge in our approach is the scarcity of data. Although it is always possible, in principle, to acquire more experimental data, in practice the amount of data is unlikely to change because the cost is too high. In the laser pattern case, one possibility to circumvent this limitation is to combine data from experiments on several materials using domain adaptation [Red+19], which would increase data by an order of magnitude and open the door to exploring patterns in unseen materials, which has great interest for applications.

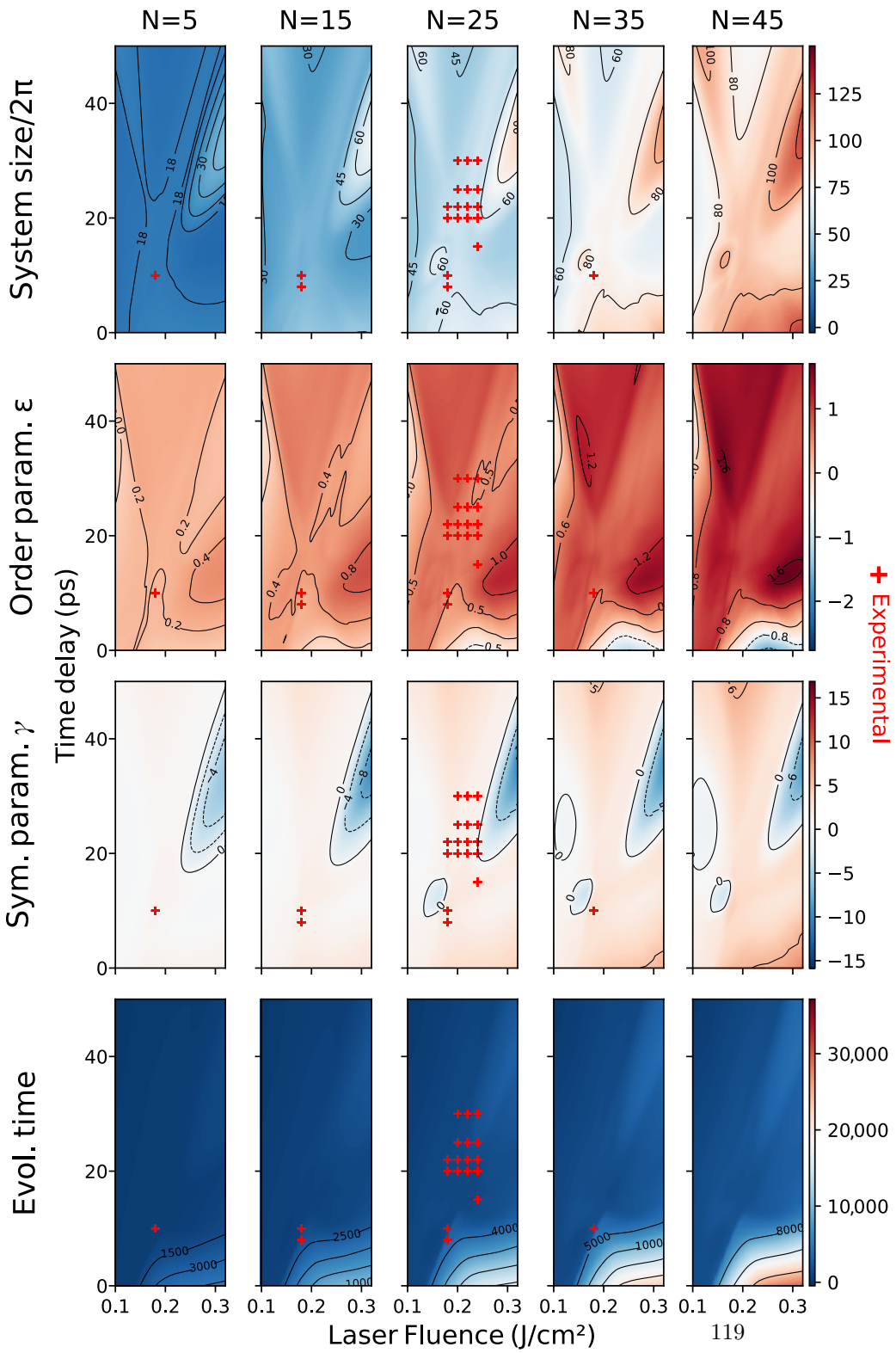


Figure 3.24: Each plot shows the predictions of the direct model, trained on the "full" dataset, of a single SH parameter (bottom to top: system size in multiples of 2π; order parameter ϵ; symmetry breaking parameter γ; simulation evolution time) as a function of laser fluence, time delay, and number of pulses (respectively, x-axis and y-axis, and column). Experimental points are overlaid on each plot.

Chapter 4

Is my Neural Net Driven by the MDL principle?

4.1 Introduction

New data often traces out regularities found in past observations, an idea known as generalization: finding regularities that are consistent with available data which also apply to data that we are yet to encounter. In the context of supervised machine learning we measure it by *learning* the rules on observations by minimizing some loss function, and evaluating it on observed and unobserved data. The difference between risk in the training data and new observations is known as the *generalization gap*. When it is small, the model generalizes well.

In the context of empirical risk minimization the generalization gap can be estimated in terms of model complexity, which increases with its number of parameters. We thus expect to reduce the generalization gap through a form of *regularization*, either by explicitely reducing the number of parameters, controlling a norm [KH92; YM17], or e.g. using dropout [Sri+14; HL15] or batch normalization [IS15; Luo+18; San+18].

Surprisingly, neural networks (NN) trained by stochastic gradient descent (SGD) generalize well despite possessing a higher number of parameters than training data, even without explicit regularization [GBC16]. An elegant explanation for this phenomenon is that SGD implicitly controls model complexity during learning [NTS14; HRS16], resulting in networks that are significantly simpler than their number of parameters suggests, as shown by several metrics to assess effective capacity, e.g. the model's number of degrees of freedom [GJ16], which is related to generalization gap, or its intrinsic dimension [Li+]. It is thus puzzling that, in spite of their implicit simplicity, NN classifiers trained by SGD are able to perfectly fit pure random noise [Zha+21], even while explicitly using regularization. In pure random noise, there is no signal to learn a rule from, and to reduce the generalization gap we must reduce the *training* performance. Since common regularization methods are unable to achieve this, using them to control model expressiveness does not address generalization: we need to "rethink generalization".

To do so we offer the following insight. To learn, from noisy observations, regularities that apply to data that we are yet to encounter, we must do so in a noise insensitive way: we must learn from signal rather than from noise. If we do so, there is no generalization gap when learning from pure

noise: since there is no signal, the model would simply not learn at all!

In this chapter, we shall give a formulation of this insight in terms of a minimum description length principle (MDL), [Ris78; Ris83] a principle of model selection which can be seen as a formalization of Occam's Razor. MDL states the problem of learning from data in terms of finding regularities that we can use to compress it: *choose the model that provides the shortest description of data, comprising the model itself*[1]. This idea was formulated in different ways since it was first advanced in [Ris78], to respond to technical difficulties in application [Grü05]. In the original, two-part form, restricting the model class to finite sets, application of this principle turns into Kolmogorov's minimal sufficient statistic [VL00].

MDL expresses the ability to generalize in terms of compressibility, which can be motivated using three main facts: (i) regularities in a random variable X can be used to losslessly compress it (ii) the minimum achievable code length is the entropy (iii) it is very unlikely that data that has no regularities can be compressed. Taken together, these imply a model's ability to compress data is likely due to finding a regularity, which will likely be found in new data as well. It is this intuitive appeal that motivates the use of MDL in spite of some conceptual difficulties, namely in selecting the encoding used to measure the length of the description of the model, which depends on the choice of encoding.

To address this difficulty, we propose an approach that uses both the signal and the noise in the data to implicitly define model complexity unambiguously: *Choose the model whose representation of the data can be used to compress the signal, but not the noise.*

Formalizing this statement requires a perspective of signal and noise that is particularly adjusted to classification problems, where the signal is task-defined [Grü05], and everything else can be considered as noise. As we shall see, our MDL statement has a significant impact on the distribution of the singular values of the point Jacobian matrices of a NN. Networks that learn from noise (where their output can be used to compress the noise) tend to maximize singular values in arbitrary directions to capture the fake "signal" in local directions. As a result the spectrum is uniformly distributed. On the other hand, NN that learn from signal but not from noise (where their output can be used to compress the signal but not the noise) tend to capture local regularities in the signal by maximizing singular values in directions aligned with the data. These directions are, by definition of signal, not arbitrary. Since the network also tends to ignore everything that is not signal, by minimizing singular values in arbitrary directions, in the limit of infinite epochs, this results in a spectrum distributed according to a power law, with a large proportion of small singular values and a fat tail.

Our contributions Our main contributions in this chapter are 3-fold: (i) we provide a formulation of the MDL principle that is generally applicable to learned representations (ii) we provide a capacity measure based upon this principle (iii) we show experimentally that neural networks are driven by the MDL principle.

Chapter organization This chapter is organized as follows: Sec. 4.2 contextualizes of our work, focusing on the sensitivity measure provided in [Aro+18]. We then provide a few information theoretic results in Sec. 4.3.2 to contextualize our definition of signal and noise in Sec. 4.3.6. Section 4.4 is the core of our contribution: we define our MDL objective in Sec. 4.4.1, and provide the local approximation in Sec. 4.4.2 that allows us to predict the spectral distribution in Sec. 4.4.2. In Sec. 4.5

[1] This formulation is known as two-part MDL, which depending on the author can be seen as "traditional" (in opposition to "modern" MDL which uses a one-step encoding using universal encodings [Grü05]) or "pure [VL00]".

we present experimental results[2] which allow us to conclude in Sec. 4.6 that neural networks are driven by the MDL principle, and discuss future work.

4.2 Related work

MDL has traditionally been used for model selection [Ris01; HY03; Grü05; BRY98; MNP06], but its intuitive appeal has led to applications in other areas such as pattern mining [Gal22; Hu+15]. In supervised learning, MDL was used in NN as early as [HV93], in which the authors added Gaussian noise to the weights of the network to control their description length, and thus the amount of information required to communicate the NN. In classification, existing approaches are inspired in MDL for density estimation [Grü05], and most can be reduced to the same approach based on the 0/1 loss, which, while not making probabilistic assumptions about noise, was shown to behave suboptimally [GL07]. Existing modifications to address this [Bar91; Yam98] do not have, unlike our approach, a natural coding interpretation. Finding a formulation of MDL for classification that can be applied in general and realistic settings is thus an open problem, and this thesis aims to contribute in this direction.

The relationship between noise, compressibility and generalization has been explored in [BL03], for example, to derive PAC-Bayes generalization bounds, or in the information bottleneck framework [TZ15]. Closer to our approach [Mor+18a] studies the stability of the output of NN with respect to the injection of Gaussian noise at the nodes, experiments showing that networks trained on random labels are more sensitive to random noise. In [Aro+18], the notion of stability of outputs is extended to layer-wise stability, improving network compressibility and generalization. The authors define layer sensitivity with respect to noise (essentially the expected stable rank with respect to the distribution of the noise), and show that stable layers tend to attenuate *Gaussian* noise. A compression scheme is provided for the layer weights that acts on layer outputs as Gaussian noise, which subsequent stable layers will thus tend to attenuate. This, since the output of the network is unchanged, shows that a network composed of stable layers is losslessly compressible. A generalization bound for the compressed network is then derived in terms of the empirical loss of the original network and the complexity of the compressed network. This work shows a clear connection between compressibility of the model and generalization, but the connection to MDL is less evident. We will show that enforcing our MDL principle leads to a measure that can be seen as an average of local sensitivities, which are similar to those defined in [Aro+18], but with crucial differences. In our approach, sensitivity is logarithmic, direction-dependent, and importantly combines sensitivity to signal and to noise.

4.3 MDL principle, signal, and noise

In this section we offer a brief overview of the Minimum description length principle. We explain how it can be applied to the case of Neural Network classifiers, and motivate our new contribution in Section 4.4. We strove to keep the section self-contained for readers new to the field, which is vast and covers over 40 years of development.

However, the section is clearly geared towards MDL for neural network classifiers operating in the overparameterized regime. We thus put ourselves in the context of finite parametric probabilistic

[2]Repository: https://anonymous.4open.science/r/ismymodeldrivenbymdl-96BA/.

families, since this is the case of interest for this application. MDL has traditionally been used for model selection, where a "model" is actually a family comprised of different probability distributions. We focus on the parametric probabilistic family case, again since it is the case of interest for neural network applications.

In its original two-part form[Ris78], MDL states the learning problem as finding regularities in data, which is identified with 'ability to compress'. This has important philosophical appeal, as it avoids assuming a data-generating process, a 'true' from which comes the data.

This is not the case in some more recent formulations of MDL in terms of one-part codes, called *expectation-based MDL* in [Grü05]. In our presentation, we shall focus on MDL *in the individual sequence sense* where possible, to keep with the clear interpretation even when no assumptions are made with respect to a 'true' data generating process.

This section is organized as follows: in Section 4.3.1 we introduce definitions and terminology that will be use throughout the section, which we illustrate using a toy example. In Section 4.3.2 is a short information-theory primer, focusing on the Kraft-Macmillan inquality[Kra49] and the existence of optimal codes. It is intended as an intuitive motivation for the MDL approach to learning.

The MDL principle was originally introduced in a two-part form, which we present in Section 4.3.4, where we also motivate the need for one-part codes.

In Section 4.3.5 we present a modern formulation of MDL, which is based upon the idea of universal codes and the minimization of maximum regret in the individual sequence sense. We show that in a number of important applications (namely neural network classifiers), the universal distribution is the normalized maximum likelihood (NML) distribution. The latter is generally uncomputable, but provides great insight into the MDL objective. We present a few of these interpretations, focusing on Rissanen's[Ris00], which will be the basis for the subsequent arguments in this Section. We present and discuss a precise definition of noise in the MDL context in Section 4.3.6.

In the following Section4.3.7, we show how MDL can be applied to the context of neural network classifiers by framing it as a communication problem. We conclude in Section 4.3.8 with a discussion that originally shows that stochastic complexity minimization is unsuitable for model selection in this context, in the overparameterized regime, and propose a novel approach, which specifies complexity implicitly in terms of signal and noise. This approach will be formalized and applied in Sections 4.4 and 4.5 respectively, which constitute the core of our contributions.

4.3.1 Preliminaries and notation

MDL lies at the intersection of information theory, statistics. It draws terminology and notation from both and uses some of its own.

Models and model families

Denote $x^n := x_1, x_2, \ldots, x_n$ a sequence of elements taken from finite or countable sample space $\mathcal{X}$. Let p^n denote a probability distribution on $\mathcal{X}^n$, with $p^n(x^n)$ the probability of x^n and X^n the corresponding random variable, that is $p^n(x^n) = \Pr(X^n = (x_1, x_2, \ldots, x_n))$. We write $p^n(x^n) := p(x^n)$ whenever this is not a source of confusion.

Definition 15 (Probabilistic source)**.** *A* probabilistic source[3] *P is a sequence of probability dis-*

[3] Also known as *information source* in the literature.

tributions $p^1, p^2, \ldots, p^k, \ldots$ *on* $\mathcal{X}^1, \mathcal{X}^2 \ldots, \mathcal{X}^k \ldots$ *, such that for all* n*,* p^{n+1} *is* compatible *with* p^n*, that is* $p(x^n) = \sum_{y \in \mathcal{X}} p(x^n, y)$*.*

We say that data are i.i.d. under source P if for each n, we have $p(x^n) = \prod_{i=1}^{n} p(x_i)$. If $\mathcal{X}$ is continuous, the sum in the compatibility condition is replaced by an integral.

Definition 16 (Probabilistic model)**.** *A probabilistic model* $\mathcal{M}$ *is a set of probabilistic sources. We usually use it to denote sources of the same functional form, which are typically indexed* $\mathcal{M}$ *with a parameter* θ *over some set* Θ*. In that case, we denote the probability mass function of the source indexed by* θ *as* $p(\cdot|\theta)$*: in particular, the probability of observing* $x^n = x_1, x_2, \ldots, x_n$ *is given by* $p^n(x^n|\theta) = p(x^n|\theta)$ *in the notation above.*

Definition 17 (Parametric family)**.** *A probabilistic model* $\mathcal{M} = \{p(\cdot|\theta) : \theta \in \Theta\}$ *is called a* parametric model *or a* parametric family *if* $\Theta \subseteq \mathbb{R}^k, k \geq 1$ *is connected and if, for all* n*, for all* $x^n \in \mathcal{X}^n$ *the mapping* $\theta \to p(\cdot|\theta)$*, viewed as a function of* θ*, is well-defined and continuous. We call the parametric family* smooth *if in addition the mapping is infinitely differentiable.*

Codes and codelength

By "coding" we mean to describe a samples or a sequence of samples from a random variable X by a symbol or sequences of symbols from an *alphabet* — a finite or countable set of *symbols*.

Definition 18 (Source code)**.** *A source code* $C(X)$ *(*C *when there is no risk of ambiguity) for random variable* X*, is a function from* $\mathcal{X}$ *the range of* X *to* $\mathcal{D}^*$ *the set of finite strings of a* d*-ary alphabet* $\mathcal{D}$*, associating* $x \in \mathcal{X}$ *to a codeword* $C(x)$*. Where it exists, the inverse of a non-singular code is called the decoding function.*

The *length* of the codeword $l(x)$ is the number of elements in $C(x)$, and the expected code length is $L(X) := \mathbb{E}_X[l(x)]$. A code is said to be *non-singular* if every $x \in \mathcal{X}$ maps to an unique element of $\mathcal{D}^*$. An *extension* C^* of code C codes sequences $x_1 x_2 \cdots x_n : x^n$ of elements of $\mathcal{X}$ as the concatenation of individual code words: $C(x^n) = C(x_1)C(x_2)\cdots C(x_n)$. The image x_n by C is called an *encoding*.

A code is said to be *uniquely encoded* if its extension is non-singular. Since in that case, every element in $\mathcal{X}^n$ is unambiguously encoded with a unique string, non-singular codes allow us to losslessly compress data.

Definition 19. *A code* $c : \mathcal{X} \to \mathcal{D}^*$ *is called a* prefix code *or* prefix-free code *or* instantaneous code *if no codeword is a prefix of any other codeword[CT12], that it, for all* $x, y \in \mathcal{X}$*, there is no* $s \in \mathcal{D}^*$ *such that* $c(x) = c(y)s$*.*

Prefix codes are uniquely encoded. In this thesis, unless otherwise stated, codes will be taken to mean 'prefix codes'.

Example 4.1. *Consider the set of Fruits* $= \{apple, mango, orange, \}$ *and a fruit salad consisting of samples from that set. We build a few different binary non-singular source codes for a random variable* X *taking values on* Fruits*,* c_i : Fruits $\to \{0,1\}^*$*.*

1. *A fixed length prefix code:* $c_1(apple) = 00, c_1(mango) = 01, c_1(orange) = 10$*.*
2. *A variable length prefix code:* $c_2(apple) = 0, c_2(mango) = 10, c_2(orange) = 110$*.*

3. A variable length non-prefix code: $c_3(apple) = 0, c_3(mango) = 00, c_3(orange) = 000$.

Encodings can be seen as messages. Consider Alice, a sender or *encoder* and Bob, a receiver or *decoder*, who is to prepare a fruit salad for dessert. They meet at a certain point in time and agree on the alphabet $\mathcal{D}$, and a source code that Alice will use to send messages to Bob. If the source code is prefix, Bob will be able to decode every message Alice sends uniquely. Using code 1, upon receiving the string 000000000110, Bob knows that he should prepare an apple, mango, orange fruit salad, with one orange and one mango, and four apples. Had they agreed to use code 2 instead, Bob would know to prepare the same fruit salad if he had received the string 000010110. Bob would be able to unambiguously decode every possible message that Alice could send: either code has a non-singular extension.

But using the non-prefix code 3, Alice would send 000000000 for the desired recipe. Bob, however, could also interpret this message as 'three oranges' and prepare that instead. The non prefix code extension is singular, and for that reason, we consider consider prefix codes only.

Finally, we note that although codes 1 and 2 are equally good to communicate unambiguously and assuredly, and the encoded message is shorter than the original one, the *length* of the encodings are different. If Alice and Bob want to communicate expediently, they could be interested in keeping the code length as small as possible. In that case, if Bob really likes apples in his fruit salad, it would be reasonable to use the variable length prefix code c_2 rather than the fixed-length one c_1: apples are encoded using a single bit in the former, and use up two bits in the latter. Messages from Alice would tend to have lots more apple in them than oranges and mangoes. If a short encoding in expectation is desirable, it is thus reasonable to pick short codewords to encode apples rather than the less frequent oranges and mangoes.

This statement about Bob's preferences induces a "regularity" in data, and this regularity can be used to the advantage of the code designer to minimize expected code length: assign short codewords to frequent outcomes and keep the longer codewords for the least frequent outcomes.

As we shall see in 4.2, there is a limit to how much we can minimize this expected code length: the Shannon entropy of random variable X, "Alice's choice of fruit for the fruit salad".

As we discussed in Section 2.3.3, given a random variable X which encodes knowledge about the likelihood of outcomes, its Shannon entropy $H(X)$ measures the expected surprise when sampling from that distribution. If Alice has "a lot of knowledge" about Bob's fruit preferences, she will assign large probability to his favorite fruit, which results in a peaky distribution, and small entropy. She can use this knowledge to design a code that is short in expectation.

If on the other hand she knows nothing about Bob's fruit preferences, she has no reason to assign a greater probability to either fruit. This will result in a uniform distribution with high entropy and long expected minimal codelength.

4.3.2 A few fundamental results in Information theory

MDL can be motivated by three fundamental information-theoretic results: (i) regularities in a random variable X can be used to losslessly compress it using a non-singular code for X; (ii) the minimum achievable codelength is the entropy; and (iii) it is extremely unlikely that data that has no regularities can be compressed. In this section, we motivate (i) in 4.3.2 with a toy example, prove (ii) and provide an informal argument for (iii) (see e.g. [CT12] or [MM03] for proofs in more general settings). A similar argument can be used to prove a finite-precision version of the Theorem 1 in [Zha+17], which provides a necessary condition for a 2-Layer ReLU network to be able to perfectly fit the training data. A straightforward application of this original result allows

us to show 4.3.3, for example, that a two-layer network that can be losslessly compressed to less than about 125 kB cannot perfectly overfit cifar-10 [KH+09].

Using regularities to compress To motivate (i), consider an object of mass m falling freely from a height h_0 on Earth (acceleration of gravity g), and a table recording heights $\{h_1, h_2, \ldots\}$ at times $\{t_1, t_2, \ldots\}$. which are known to obey $h(t) = h_0 - \frac{1}{2}mgt^2$ since Galileo. We do not expect this law to hold exactly: it is clearly an *approximation*, assuming negligible air resistance, uniform gravitational field, etc. But we do expect it to be *good*, in the sense that the better the conditions of the approximation hold, the more can deviations to the results predicted by Galileo's law be seen as corrections.

This is the key to understanding how the regularity captured by Galileo's formula can be used to losslessly compress the height-times table. Since we expect Galileo's formula to predict the first significant digits of the height with high confidence, and measurements are performed and stored with finite precision, instead of storing x_i, h_i, we can store $x_i, \Delta h_i = h_i - h(t_i)$. We can thus store the *same* data (in expectation) using *less* digits, which amounts to lossless compression. The more regularities we are able to find in data, the more we can compress it.

But we can do even better. Galileo's law ignores air resistance, for example. A better "law", taking e.g. drag into account (while still assuming uniform gravitational field, etc.), increases confidence in the first significant digits of the predictions, reducing in expectation the number of significant digits of the deviations, allowing better compression.

This is, however, achieved at the expense of an increase in description length of the "law" itself. Whereas with Galileo's law, we only need to store m and g, using a more accurate model we need to store additional quantities. There is a trade-off between the description lengths of the data and the model, as a better model takes longer to describe (although we expect this to saturate at some point). In the limit, a very large model can decrease the description length of finite data simply by memorizing it. Notably, two-layer ReLU feed forward NN can do this with surprising ease [Zha+17] but, as predicted in the MDL framework, at the expense of an increase in complexity [BO18].

Kraft Inequality and Optimal codelength Results (ii) and (iii) crucially rest on the Kraft-Macmillan inequality, which we state and prove below for the case of prefix codes (known as the Kraft inequality) for completeness, and then use to show (ii). See [CT12] for a more detailed treatment.

Theorem 4.1 (Kraft-Macmillan inequality)**.** *For any uniquely decodable code C over an alphabet of size D, the codeword lengths $l_1, l_2, \ldots, l_m$ must satisfy the inequality*

$$\sum_i D^{-l_i} \leq 1 \tag{4.3.1}$$

Conversely, given a set of codeword lengths that satisfy 4.3.1, there exists a uniquely decodable code with these word lengths.

Proof. The idea of the proof for prefix codes (known as the Kraft inequality) is that prefix codes from a D - adic alphabet can be seen as the childless nodes on a rooted tree. No prefix code can then be among the descendants of another. Hence, the sum of the descendants of prefix codes cannot exceed the number of leaves of the tree: $\sum_i D^{l_{max}-l_i} \leq D^{l_{max}}$. The converse is established simply by noting that lengths that satisfy 4.3.1 can be placed on rooted tree. If they couldn't there

would be more words of length l_i than descendants of non-used codes; but since the total mass at every level is constant, this cannot happen. □

As a straightforward application of Kraft's inequality, we can immediately see that all codes 1, 2 and 3 are uniquely decodable, since, respectively, $\sum_i 2^{-l_i} = 1/4 + 1/4 + 1/4 < 1$, $\sum_i 2^{-l_i} = 1/2 + 1/4 + 1/8 < 1$, and $\sum_i 2^{-l_i} = 1/2 + 1/4 + 1/8 < 1$.

Theorem 4.2 (Optimal code length)**.** *The expected code length for any uniquely decodable code C of a random variable X over an alphabet of size D is greater than or equal to $H_D(X)$ the entropy calculated in base D, with equality holding iff $D^{-l_i} = p_i$.*

Proof. To establish this, consider the difference between the entropy and the expected length:

$$\begin{aligned} H(X) - \sum_i p_i l_i &= H(X) + \sum_i p_i \log_D D^{-l_i} \\ &= -\sum_i p_i \log_D p_i + \sum_i p_i \log_D \frac{D^{-l_i}}{\sum_j D^{l_j}} + \log_D \sum_j D^{-l_j} \\ &= -KL(p||\frac{D^{-l_i}}{\sum_j D^{l_j}}) + \log_D \sum_j D^{-l_j} \\ &\leq 0 \end{aligned}$$

where the non-positivity at last step follows from the non-negativity of the Kullback-Liebler divergence (a consequence of the concavity of the logarithm, via Jensen's inequality, see [CT12]), and from the Kraft-Macmillan theorem 4.1 for the second term. To prove equality iff $D^{-l_i} = p_i$ we again appeal to the properties of the Kullback-Liebler divergence: the first term in the sum is zero iff $p_i = \frac{D^{-l_i}}{\sum_j D^{l_j}}$, in which case $\log_D \sum_j D^{-l_j} = \log_D 1 = 0$. This proves the claim. □

An optimal prefix code always exists (e.g. Huffman code), but for our purposes, the Shannon-Fano code, which sets codeword lengths $l(x) = \lceil -\log p(x) \rceil$ suffices. The Shannon-Fano code is competitive, meaning that the probability that the expected length exceeds another code's by c bits does not exceed 2^{1-c} [CT12]. Indeed for the Shannon-Fano code, the Kraft-Macmillan inequality 4.1 is automatically satisfied and by definition, $l(x) < -\log p(x) + 1$. Hence, in expectation, for any other code with expected length L' we have

$$\begin{aligned} L &< -\sum_x p(x) \log p(x) + 1 \\ &= H(X) + 1 \\ &\leq L' + 1 \end{aligned}$$

Incompressible data Finally, to justify (iii) the statement that it is extremely unlikely that data with no regularities (with maximal entropy) can be compressed, we provide the following argument. By the Kraft-Macmillan inequality 4.1, for every prefix code of a random variable X over an alphabet of size D, the expected codeword length is no greater than the entropy, with equality iff the $l_i = -\log_D p_i$. Assuming X is discrete, all n events have the same probability $\frac{1}{n}$.

Hence, the expected code length (per symbol) is $L \geq -\sum_{i=1}^{n} p_i \log_D p_i = \log_D n$. The lower bound is what we can achieve simply by assigning each codeword to the leaves of a D-nary tree: the best code coincides with the worst possible code, and so data cannot be compressed.

4.3.3 Finding a finite precision network that overfits

Memorizing data with Neural Networks Two-layer ReLU feed forward neural networks are able to memorize data for any given, arbitrary sample size n, data in arbitrary dimension d with surprising ease: as stated in [Zha+17], Theorem 1, within the set of such networks $\mathcal{N}$ with at least $2n + d$ parameters, there is at least one $N \in \mathcal{N}$ that will be able to compress y perfectly, by expressing it in terms of x. This is *not* at odds with the MDL principle (i) since MDL states that compressibility of data without a rule is *unlikely*, rather than impossible (ii) the description lenght of the network is not taken into account (see [Mor+18a]).

The proof of Theorem 1 in [Zha+17] implicitly assumes infinite precision in the weights of the network, which would take up infinite, and thus unavailable, space. For completeness, we briefly restate it and provide a gist of the proof in [Zha+17]. We then use this intuition to prove a simple result showing how overfitting in finite precision (i.e. with real data and real networks) imposes constraints on the *size* of the network, not just the number of parameters.

Finally, we apply this new result to cifar-10, showing that with 32 bit precision, $2n+d$ parameters are far from enough to overfit cifar-10-like data.

Theorem 4.3 ([Zha+17], Theorem 1)**.** *There exists a two-layer neural network C with ReLU activations and $2n+d$ weights that can* represent *any function on a sample of size n in d dimensions, in the sense that, for every sample $S \subseteq \mathbb{R}^n$ with $|S| = n$ and for every function $f : S \to \mathbb{R}$, there exists a setting of the weights of the C such that $C(x) = f(x)$, for all $x \in S$.*

Proof gist. The proof rests on a Lemma that constructs a matrix A that is lower-triangular and has non-zero and distinct real diagonal elements: the first differences of an increasing sequence. A is hence non-singular, since the diagonal elements of a triangular matrix are its singular values. The authors then proceed to stating the overfitting problem for a $2n + d$-parameter 2-layer ReLU network $c(x) = \sum_{j=1}^{n} w_j \max\left\{a^\top x - b_j\right\}$ with weight vectors $w, b \in \mathbb{R}^n$ and $a \in \mathbb{R}^d$ as the solution of a linear system in a matrix B. This matrix can be made of type A via a judicious choice of network parameters a and b. Precisely, one needs to chose a such that for every sample x_j we have $a^\top x_j < a^\top x_{j+1}$. This can always be done for distinct x_i, by the Archimedian property of the reals. It remains to select b_j such that $a^\top x_j < b_j < a^\top x_{j+1}$, which can be done because $\mathbb{R}$ is complete. The remaining parameters of the network w are precisely the solutions of the system in B. □

Memorizing data with finite-precision Neural Networks In finite precision, it is not always possible to find a, b, w as in the theorem above. The least number of bits per parameter for this to happen is given by the following proposition.

Proposition 4.1. *Let data be composed of inputs $\{x_1, x_2, \ldots, x_n\} \subseteq \mathbb{R}^d$ and labels $\{y_1, y_2, \ldots, y_n\} \subseteq \mathbb{R}$, with respectively $s_x > s_y$ denoting the least significant figures of the least significant component of, respectively the inputs, and the labels: $s_x := \min_{i,j} \left\{x_j^k\right\}$ and similarly for the inputs.*

Then to be able to represent this data it suffices a neural network classifier with d parameters with the same significant digits than x, n parameters with the one more significant digits than x,

and n parameters with the same number of significant digits as y, For an expected number of bits of $(s_x d + n(s_x + 1) + ns_y) \log_2 10$.

Proof. Each $a^\top x_j$ has at most the number of significant figures as the $a^k x_j^k$ with the least significant figures, which is s_x. Hence, each a^k has at most s_x significant figures, for a total of $s_x d$ for all components.

The problem of finding constant b such that we can place a b between every two $a^\top x_j$ can be done assuredly by picking each of the n parameters b with one more significant figure per component than x_j, so $n(s_x + 1)$. This allows us to construct the non-singular matrix $A = [a_{ij}] = [\max\{x_i - b_j, 0\}]$ as in the proof of 4.3. Given A, the w are solutions of a system $y = Aw$. In classification, y is the limiting term, which implies that each of the n components of w has the number of significant figures of y, so ns_y. The total number of significant base 10 digits is therefore $K = s_x d + n(s_x + 1) + ns_y$. Assuming all of them are independent, these can be encoded in $K \log_2 10$ bits, which proves the claim. □

Memorizing cifar-10-like data We now apply this new result to cifar-10-like data in the setting of a typical Machine Learning pipeline, where the number of significant figures is fixed. For simplicity, we assume that the precision of the data and weights are the same: a 32 bit float.

The result gives an upper bound to the parametric complexity of the model family '2-layer ReLU networks in finite precision' $\mathcal{M}_{\mathrm{ReLU}-2}$: we will be able to perfectly fit with $m \in \mathcal{M}_{\mathrm{ReLU}-2}$ data with precision s_x, s_y with $2n + d$ parameters as long as m satisfies the constraints in Prop 4.1.

Example 4.2 (Overfitting cifar-10 with 2-layer ReLU network)**.** *The dataset cifar-10 is composed of $n = 6 \times 10^4$ 8 bit square images with 32 pixels size and 3 channels, each image belonging to one of 10 classes. Hence, $n = 6 \times 10^4, d = 3 \times 32^2$ and in bits, $s_x = 8$ and $s_y = \log_2 10$. Hence, in bits, $s_x d + nd(s_x + 1) + ns_y \approx 95.5\,\mathrm{KB}$, which is significantly smaller than 492.3 KB required to store the $2n + d$ parameters at 32 bit precision. Note that this is the size of the network* before *fitting the data. The weights after fitting cifar-10, for example, could be compressed at a much higher rate.*

4.3.4 From two-part to one-part MDL

MDL was devised as an extension of Kolmogorov's algorithmic complexity which, we have seen before, in spite of its intuitive appeal, suffers from the important fact that it is fundamentally immeasurable.

Crucially, Kolmogorov complexity measures complexity without appealing to a "true" underlying generating process, but suffers from the fact that it is uncomputable.

To solve this issue, Rissanen proposed to, rather than state complexity in terms of a universal computer, to focus on the description length with respect to a given model family instead. Introduced by Rissanen in [Ris86] as a modification of Kolmogorov complexity, *Stochastic complexity* is the shortest description length of data within that model family (see [Ris97] for an exposition and results) .

Definition 20 (Stochastic complexity)**.** *Given a probabilistic model family $\mathcal{M}$ and data $\mathcal{D}$, the* stochastic complexity *of $\mathcal{D}$ with respect to model family $\mathcal{M}$* $\mathbf{SC}(\mathcal{D}, \mathcal{M})$ *is the shortest possible code length that can be obtained by using models in $\mathcal{M}$.*

Equating learning with compression, the code that provides maximal compression within a model family $\mathcal{M}$ allows us to obtain "all the useful information in observed data that can be extracted with selected class of modeled distributions" [Ris87].

As should be clear from Sec. 4.3.2, a principle that equates learning with compression must also take into account the size of the compression method itself: using the free-fall data table as example, a model that is sufficiently complex to memorize the table in full is able to perfectly compress it, but at the expense of having enough complexity to memorize it.

This motivates the original formulation of MDL as a principle of model selection, which we call *two-part MDL* following [Grü05]:

Definition 21 (Two-part MDL)**.** *Let $\mathcal{M}^{(1)}, \mathcal{M}^{(1)}, \ldots, \mathcal{M}^{(n)}$ a set of candidate models (i.e. model* families*) to explain data $\mathcal{D}$. Then the best candidate model (i.e. distribution) $m^* \in \bigcup_i \mathcal{M}^{(i)}$ is the one which minimizes the sum*

$$m^* = \underset{m \in \bigcup_i \mathcal{M}^{(i)}}{\arg\min} \; L(m) + L(\mathcal{D}|m)$$

where $L(m)$ is the description length of m and $L(\mathcal{D}|m)$ is the description length of the data when using the Shannon-Fano encoding corresponding to m. The best model family $\mathcal{M}^*$ to explain $\mathcal{D}$ is the one that contains m.

Put differently, the best model family $\mathcal{M}^*$ is the one minimizing stochastic complexity $\mathbf{SC}(\mathcal{D}, \mathcal{M}^*)$, and the best model is the one for which this code length is achieved[4]

In terms of the taxonomy in Sec. 2.2, MDL measures complexity in the sense of difficulty of description. In spite of its intuitive appeal, this formulation has a fundamental problem. The second term measures description length with respect to the model m. As we use one of the distributions in $\mathcal{M}$ to create an optimal encoding (Shannon-Fano for definiteness), it is well-defined. As for the first term $L(m)$, however, there is no natural encoding with respect to which to measure the description length of m. In this sense, the description length of the data given the model and of the model itself are expressed in 'different units'.

In the next section, we present a one-part formulation that solves this problem. Unfortunately, as we shall see, it also introduces new ones.

4.3.5 One-part MDL

The basic idea of one-part MDL is to measure complexity with respect to a representative of the model family that compresses data well in the worst case.

This representative, which we will define rigorously below, is called a *universal model*[5]. We then show how we can formulate model selection as a minmax problem by presenting a definition of *regret*. For parametric model families and under regularity assumptions, the solution to this minmax problem is called the *normalized maximum likelihood* distribution.

Using this solution opens up a number of insights with respect to the nature of the MDL solution, and importantly with respect to the model complexity term.

Basic idea: universal codes

The best code for data x^n in a parametric model family $\mathcal{M} = \{p(\cdots|\theta) : \theta \in \Theta\}$ is given by the maximum likelihood estimator for x^n, which has length $-\log p(x^n|\hat{\theta}(x^n))$. Unfortunately, $p(x^n|\hat{\theta}(x^n))$ is *not* a distribution, since $\sum_{x^n} p(x^n|\hat{\theta}(x^n)) = \sum_{x^n} \max_{\theta \in \Theta} p(x^n|\theta)$. If Θ has more than one element,

[4] The statement is simplified, as in general more than one minimizer can exist.

[5] But this time, in the sense of a distribution belonging to a model family.

there is always at least one x^n for which the conditional probabilities are different, and if this set has nonzero measure, the total mass therefore exceeds one.

We could still use $p(x^n|\hat{\theta}(x^n))$, of course, but that would require us to either know the maximum likelihood estimator $\hat{\theta}(x^n)$ in advance (which is why it is called "best code in hindsight"), or else to encode and send the parameters $\hat{\theta}(x^n)$, which would be equivalent to a two-part code and thus raise the same problems (how do we encode them?, etc.).

Regret A possible solution to this problem is to pick a code that is *closest* to this best code in hindsight. We measure closeness using the notion of *regret*:

Definition 22 (Regret)**.** *Let $\mathcal{M}$ be a probabilistic model and p a distribution on $\mathcal{X}^n$ (not necessarily in $\mathcal{M}$). Then for given $x^n \in \mathcal{X}^n$ the* regret *of p relative to $\mathcal{M}$ is defined as*

$$\mathcal{R}\left[p, \mathcal{M}\right](x^n) := -\log p(x^n) - \min_{q \in \mathcal{M}} \{-\log q(x^n)\} \tag{4.3.2}$$

If the maximum likelihood estimator $\hat{\theta}(x^n)$ is a function, then

$$\mathcal{R}\left[p, \mathcal{M}\right](x^n) = -\log p(x^n) + \log p(x^n|\hat{\theta}(x^n)). \tag{4.3.3}$$

Regret is thus the minimum number of extra bits needed to encode x^n using elements of $\mathcal{M}$ compared to the optimal code in hindsight (the code that we could use would we have known the data to do maximum likelihood, or the maximum likelihood estimator itself). Note that since we allow p to be outside $\mathcal{M}$, regret can be negative.

Optimal codes minimize worst-case regret We thus define worst-case regret, which will be used as a measure of distance to the distribution in $\mathcal{M}$ that would minimize description length in hindsight.

Definition 23 (Worst-case regret)**.**

$$\mathcal{R}^n\left[p, \mathcal{M}\right] := \max_{x^n \in \mathcal{X}^n} \{\mathcal{R}\left[p, \mathcal{M}\right](x^n)\} \tag{4.3.4}$$

Note that to measure Shannon entropy, instead of the maximum we look at the mean. In this case, we measure regret as the maximum extra bits with respect to all sequences $x^n \in \mathcal{X}^n$.

We define $\hat{p}$ as the optimal Universal model relative to a Probabilistic model if it minimizes worst-case regret.

Definition 24 (Optimal Universal Model)**.** *We call $\hat{p}$ the* optimal universal model *relative to a probabilistic model $\mathcal{M}$ if, denoting $P(\mathcal{X}^n)$ the set of possibly defective distributions on $\mathcal{X}^n$,*

$$\hat{p} = \arg\min_{p \in P(\mathcal{X}^n)} \mathcal{R}^n[p, \mathcal{M}] \tag{4.3.5}$$

If $\mathcal{M}$ is a parametric probabilistic family, then the optimal universal model is

$$\hat{p} = \arg\min_{p \in P(\mathcal{X}^n)} \mathcal{R}^n[p, \mathcal{M}] = \arg\min_{p \in P(\mathcal{X}^n)} \max_{x^n \in \mathcal{X}^n} \left\{-\log p(x^n) + \log p(x^n|\hat{\theta}(x^n))\right\} \tag{4.3.6}$$

Finally, we define a logarithm measure of the total mass under the curve (recall, not a distribution) that picks the maximum over all distributions in $\mathcal{M}$ of the probabilities of observing each x^n:

Definition 25 (Parametric complexity)**.** *We define the* parametric complexity *or* model cost *of* $\mathcal{M}$ *[Grü05] as*

$$\mathbf{COMP}_n(\mathcal{M}) = \log \sum_{x^n \in \mathcal{X}^n} p(x^n|\hat{\theta}(x^n)) \tag{4.3.7}$$

Implementation: Normalized Maximum likelihood

It turns out that in the case of parametric model family $\mathcal{M}$ and given some regularity assumptions, the so-called *normalized maximum likelihood distribution* has minimal worst case regret with respect to $\mathcal{M}$. This is a theorem proved by Shtarkov[Sht87]:

Theorem 4.4 (Shtarkov's theorem)**.** *Suppose that $\mathcal{M}$ is a parametric probabilistic family, and the parametric complexity $\mathbf{COMP}_n(\mathcal{M})$ is finite. Then (4.3.6) is achieved uniquely for the distribution p_{nml} given by*

$$p_{\text{nml}}(x^n) = \frac{p(x^n|\hat{\theta}(x^n))}{\sum_{y^n \in \mathcal{X}^n} p(y^n|\hat{\theta}(y^n))} \tag{4.3.8}$$

The distribution p_{nml} is called the Shtarkov distribution *or the* normalized maximum likelihood distribution *(NML).*

We adapt the proof from [Grü05] for completeness.

Proof. By definition of worst-case regret, p_{nml}, and model cost, we have:

$$\begin{aligned}
\mathcal{R}^n\left[p_{\text{nml}}, \mathcal{M}\right](x^n) &= -\log p_{\text{nml}}(x^n) + \log p(x^n|\hat{\theta}(x^n)) \\
&= -\log p(x^n|\hat{\theta}(x^n)) + \log \left(\sum_{y \in \mathcal{X}^n} \log p(y^n|\hat{\theta}(y^n)) \right) + \log p(x^n|\hat{\theta}(x^n)) \\
&= \log \left(\sum_{y \in \mathcal{X}^n} \log p(y^n|\hat{\theta}(y^n)) \right) \\
&= \mathbf{COMP}_n\left(\mathcal{M}\right),
\end{aligned}$$

which is independent of x^n. Now since for every $p \neq p_{\text{nml}}$, $\exists z^n \in \mathcal{X}^n : p(z^n) < p_{\text{nml}}(z^n)$, we have

$$\begin{aligned}
\mathcal{R}^n\left[p, \mathcal{M}\right] &= \max_{x \in \mathcal{X}^n} \mathcal{R}\left[p, \mathcal{M}\right](x^n) \\
&\geq \mathcal{R}^n\left[p, \mathcal{M}\right](z^n) \\
&\geq \mathcal{R}^n\left[p_{\text{nml}}, \mathcal{M}\right](z^n)
\end{aligned}$$

which, since the regret for p_{nml} is independent of z^n, equals the worst-case regret, which proves the claim. □

Note that the proof above also shows that, when model cost is finite, we can *define* it 4.3.7 as the regret achieved by the p_{nml} distribution for any data x^n.

Interpreting Stochastic complexity via Normalized maximum likelihood encoding

As we have just shown, by encoding data $\mathcal{D}$ with the NML distribution we obtain, by definition, a description length that exceeds the best description length in hindsight by a term that equals exactly the model complexity:

$$-\log p_{\text{nml}}(\mathcal{D}|\mathcal{M}) = -\log p(\mathcal{D}|\hat{\theta}(\mathcal{D})) + \mathbf{COMP}_n(\mathcal{M}) \tag{4.3.9}$$

As the NML distribution minimizes the worst-case risk, the left hand-side of this expression is, by definition, the stochastic complexity. By expressing stochastic complexity via the NML distribution we will provide an interpretation in terms of model complexity and of noise.

Stochastic complexity measures the relative likelihood of observed data Stochastic complexity can be thought of as the amount of information in the data $\mathcal{D}$ relative to $\mathcal{M}$. To see this note that if the normalized maximum likelihood of the data is high, then the likelihood of data $\mathcal{D}$ according to its maximum likelihood model is high relative to the sum of the likelihoods of all other data according to their respective maximum likelihood models: in terms of the model family, data $\mathcal{D}$ is likely compared to all other possible data. An encoding of $\mathcal{D}$ with elements of $\mathcal{M}$ exists such that the negative log-likelihood, the code length, is small. According to the MDL formalism, this means that the model family captures a a great deal of regularity about the data.

Stochastic complexity interpretation in terms of noise and model cost The expression (4.3.9) consists of two terms; the first term, model cost, which informally speaking, measures how likely the model family finds arbitrary data. If on average, for arbitrary x^n, there is a distribution in $\mathcal{M}$ that finds it likely, then the term will be high.

Model cost uses the behavior of distributions within the model family with respect to *data* to measure model complexity, thus solving the difficulty of choosing an appropriate measure of complexity for the model. Unfortunately, it is often infinite and generally incomputable [Grü05]. The first difficulty is commonly addressed via a *luckiness function* which essentially consists in imposing a prior on $\mathcal{X}^n$ effectively removing the diverging parts.

The first term is the shortest description length of data $\mathcal{D}$ that can be obtained within the model family. If there is an element in $\mathcal{M}$ that is *certain* about $\mathcal{D}$ then $-\log p(\mathcal{D}|\hat{\theta}(\mathcal{D})) = 0$[6].

If the the shortest description length that can be given of $\mathcal{D}$ within model family $\mathcal{M}$ is large, then there is no model in $\mathcal{M}$ that can be used to compress $\mathcal{D}$: in other words, as far as elements of $\mathcal{M}$ are concerned, observation $\mathcal{D}$ is noise.

Toy examples: empirical and Gaussian distribution model families To gain intuition, suppose that $\{x_n\} = \mathcal{X}$ is a finite set of observations of some physical experiment. Consider a model family consisting of the set of empirical distributions corresponding to these observation, i.e.

[6]This of course implies that the distribution $p(\cdot|\hat{\theta}(\mathcal{D}))$ is zero everywhere else but has no implication regarding the likelihood that *other* elements in $\mathcal{M}$ assign to other data.

$\mathcal{M} = \{\delta(x - x_n)\}$, where $x, x_n \in \mathcal{X} \subseteq \mathbb{R}$. Then for every observation $x_o \in \mathcal{X}$, since $\delta(x - x_o) \in \mathcal{M}$ and $\delta(x_o - x_o) = 1$, the stochastic complexity is

$$-\log p_{\text{nml}}(x_o|\mathcal{M}) = 0 + \log \sum_{x_o \in \mathcal{X}} 1 = \log |\mathcal{X}|$$

But suppose now that upon performing the experiment, we make some *new* observation x_{new} that was not in the original set $\mathcal{X}$. Since this new observation has zero probability for every distribution in $\mathcal{M}$, the first term in stochastic complexity is infinite.

This phenomenon is of course not specific to empirical distribution model families: observing any data that is assigned zero probability by all members of $\mathcal{M}$ is always a *infinite* stochastic complexity event. And the divergence does not mean that stochastic complexity breaks down as a measure of complexity for new data. A straightforward interpretation in terms of MDL is that the model family $\mathcal{M}$ is simply not appropriate to explain x_{new}: since MDL prescribes selecting the model family minimizing stochastic complexity, $\mathcal{M}$ would *never* be selected to explain x_{new}.

To avoid divergences, we should consider only model families for which the union of the support of its elements is the entire real line. This can easily be done, for example, by adding some infinite support distribution to the original model family, which we can assume without loss of generality is the Gaussian distribution centered on μ the mean of the observations in $\mathcal{X}$: $\mathcal{M}(\mu) = \mathcal{M} \cup \mathcal{N}(\mu)$. The stochastic complexity of the model family is now always finite:

$$-\log p_{\text{nml}}(x_o|\mathcal{M}(\mu)) = \begin{cases} 0 + \log |\mathcal{X}| + 1, & \text{if } x_o \in \mathcal{X} \\ (x_o - \mu)^2 + \log |\mathcal{X}| + 1, & \text{otherwise} \end{cases} .$$

If now one observes x_o and wishes to select amongst a set of families $\{\mathcal{M}(\mu)\}$, MDL prescribes selecting the model family such that x_o is closest to the mean of the previous observations[7]. The simplest MDL Gaussian explanation of an observation is the one whose mean is closest to the observation. This reasoning extends straightforwardly for Gaussian mixture families.

4.3.6 Signal and noise in MDL

In this chapter, we introduce an MDL principle that specifies the encoding scheme in which to measure the description length implicitly in terms of the signal and the noise in noisy data. To define signal and noise, we rely on [Ris00] which defines noise as the part of the data that cannot be compressed with the models considered, the rest defining the information bearing signal. This idea is used in the paper in the context of Gaussian models arising in linear-quadratic regression problems to derive a decomposition of data that is similar to Kolmogorov's sufficient statistics [CT12]. In our case, we shall assume that the signal is implicitly provided by a given classification task, and define noise to be everything else.

Definition 26. *We define noise as "noise relative to a signal": given r.v.s X (signal) and Δ (noise) such that $X + \Delta$ is well-defined, we say that Δ is noise relative to X if for every $C_i \in \mathcal{C}$ non-singular code of X, we have $L(C_i(\Delta)) \geq H(\Delta) + \alpha$, with $\alpha > 0$.*

[7] Note that it is irrelevant how the the mean was constructed: we did so in a reasonable way using the previous observations, but the result would stand regardless.

Note that if $C_j \in \mathcal{C}$ were optimal for Δ, then $L(C_i(\Delta)) = H(\Delta) \geq H(\Delta) + \alpha$, which with $\alpha > 0$ is a contradiction. The definition is thus equivalent to stating that there is no code of X in $\mathcal{C}$ (which may include the optimal code for X) that is optimal for Δ. Also note that the noise Δ is not particularly "disordered". Going back to 4.3.2, the physical laws that compress height vs. time data are unable to compress the effect of hitting the object with a baseball bat. Even if a model provides a simple description of some data, adding noise as defined in Def. 26 destroys its ability to compress it. It is implicit in the MDL principle that not only do we learn the regularities in data, but also the "irregularities"!

4.3.7 MDL for Deep neural network classifiers

In this section we discuss MDL in the context of classification.

Encoding labels with neural network classifiers

Suppose Alice wishes to send Bob a set of labels $y^n \in \mathcal{Y}^n$ using the least amount of bits. Rather than choosing a method that directly encodes the labels, they decide to proceed as follows. First, they select a set of publicly available data $x^n \in \mathcal{X}^n$[8] — say a set of cat pictures on Wikipedia for concreteness. Alice then assigns a label to each cat picture using a process of her choice, in order to create a dataset $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} := \{z_i\}_{i=1\ldots n}$. She then uses $\mathcal{D}$ to train a Neural Network classifier $f_\theta : \mathcal{X} \to \mathcal{Y}$ parametrized by $\theta \in \Theta$ to predict the labels, by using some flavor of gradient descent to find the $\theta(\mathcal{D}) \in \Theta$ that minimizes a classification loss function (which, we shall assume for concreteness, is cross entropy; we shall also assume that the optimization method is fixed).

Bob has access to the data x^n but he does not know the labeling procedure that Alice used. But he does not need to: if he somehow gains access to the trained neural network, he will be able to use it to predict the labels y^n from the publicly available x^n and obtain Alice's message.

Encoding labels with a neural network can be efficient Convoluted as it may seem, this method can be very efficient. Using standard integer encoding for n labels with $|\mathcal{Y}|$ label choices, sending over the labels directly takes $n \log |\mathcal{Y}|$ bits. In order to make sure that Bob has access to the trained network, Alice needs to, either implicitly or explicitely, communicate its parameters $\theta(\mathcal{D})$ to Bob.

The parameters $\theta(\mathcal{D})$ are stored with some precision, say as a 32-bit single precision float. But in general, Alice does not need to communicate the full parameters to Bob: she just needs to send them to Bob with enough precision that he can use them to select a network that will predict the same labels as $f_{\theta(\mathcal{D})}$. Assuming for simplicity that $\Theta \subseteq \mathbb{R}^n$ and that N different parameter choices for each parameter suffice to select an appropriate model, Alice could send over a $\tilde{\theta} : f_{\tilde{\theta}}(x^n) = f_{\theta(\mathcal{D})}(x^n)$ with $k \log N$ bits, which would be enough for Bob to be able to decode the labels y^n. Hence, as long as the size of the dataset is $n > \frac{k \log N}{\log |\mathcal{Y}|}$, this encoding procedure will be more efficient (i.e. shorter) than just sending over the labels. If there is a binary network that suffices for encoding cifar-10, for example, this would mean that for this encoding to be more efficient than simply sending over the labels, we would need to have approximately $k < 30n$. To be clear, at the time of this writing, the state of the art *binary* network for cifar-10 is a $k = 10$ million parameter network f with a

[8] Strictly speaking, the public data needs to be at least as diverse, as measured by Shannon entropy, as the set of labels she wishes to send over to Bob, and of the same size as the set of labels.

C10 accuracy at *test time* of 95.5[BMT20]. Assuming that f is able to reach a perfect train time accuracy, it could be used by Alice and Bob to encode cifar-10 training data. Moreover, with $n = 6 \times 10^5$, it is able to do so efficiently, because $k = 10^7$ is smaller than $n \times 30 = 18 \times 10^6$.

Train and test-time encodings Judging by its test time accuracy, it may seem that the trained network that Alice sent Bob is close to being an efficient encoder, not just for cifar-10 training data, but for all *cifar-10-like* data as well!

Unfortunately, this is not necessarily the case. As stated above, Alice sends Bob a lower-precision version of the parameters such that $\tilde{\theta} : f_{\tilde{\theta}}(x^n) = f_{\theta(\mathcal{D})}(x^n)$, amounting to $k \log N$ bits. And while $f_{\tilde{\theta}}(x^n)$ may have enough precision to decode x^n, it will not be able to do so in general. To see this, take the simple case where the network is simply a diagonal matrix for a binary classification problem $W := \begin{bmatrix} a_1 & a_2 & \dots & a_n \\ b_1 & b_2 & \dots & b_n \end{bmatrix} : \mathbb{R}^n \to \mathbb{R}^2$ trained on a single example, say $\mathbf{x}$ with label 0 for simplicity. Then all we need to do be able to decode the training example correctly is to make sure that

$$(W\mathbf{x})_0 < (W\mathbf{x})_1 \Leftrightarrow (\mathbf{a} - \mathbf{b})^\top \mathbf{x} < 0 \Leftrightarrow \cos(\theta) < 0$$

Choosing $\mathbf{b} := 0$, we just require a row vector $\mathbf{a}$ that makes an angle θ with x that is between $\frac{\pi}{2}$ and π — any row vector. In particular, we can pick a row vector with a single non-zero component, say the first, a_1. So the condition becomes $a_1 x_1 < 0$, which can be achieved by setting the sign of a_1 to the negative of the sign of x_1[9]. This low-precision classifier is able to decode the training example $\mathbf{x}$. But it would not work for any other example $\mathbf{x}'$ with label 0 such that $x'_1 a_1 > 0$, for example.

Of course if there is a finite training data/network parameter size that would allow us to reach zero *test* error for arbitrary cifar-10 like data, then the method becomes again more efficient than just sending over the labels, as Alice would only have to send a finite number of bits over to Bob, with which he could decode arbitrary-size cifar-10 like data.

Neural networks classifier architectures as probabilistic model families

Typically in this context, a neural network classifier is a probabilistic classifier[10] which, given an input $x \in \mathcal{X} \subseteq \mathbb{R}^n$, outputs the conditional probabilities of observing the labels $f_\theta(x) = p(y|x, \theta)$, with $\sum_{y \in \mathcal{Y}} p(y|x, \theta) = 1$. The classification step is performed by predicting the most probable class

$$y_{pred} = \arg\max_{y \in \mathcal{Y}} p(y|x, \theta).$$

In the language of this chapter, neural network classifier architectures (Multilayer Perceptron, VGG, etc.) can thus be seen as parametric probabilistic families (cf. Def. 17) $\mathcal{M} = \{p(\cdot|x, \theta) : x \in \mathcal{X}, \theta \in \Theta\}$. Within each model family, we select the network that we are interested in as the one for which the parameter $\theta(\mathcal{D})$ minimizes a given classification loss for the training data $\mathcal{D}$, where the minimization is performed using a flavor of gradient descent by backpropagating through the layers of the network.

[9] We assume for simplicity that we picked a component i such that $a_i x_i \neq 0$, and that this can be done.
[10] Typically, the last layer is a softmax layer of width $|\mathcal{Y}|$.

The most commonly used classification loss is the categorical cross-entropy between the empirical distribution of the labels given the inputs and the one that the network predicts for the same inputs[11]. Because, as shown in Thrm. 4.2, the length of the optimal code for the empirical distribution is the entropy, and because $\min_q H(p,q) = H(p)$, we are really looking for the distribution within $\mathcal{M}$ that has, according to the empirical distribution, the shortest code length for all inputs on average:

$$H(Y_{\text{emp}}, f_\theta(X_{\text{emp}})) = -\sum_{i=1}^{n} p_{\text{emp}}(y_i|x_i) \log p(y_i|x_i,\theta) \tag{4.3.10}$$

$$= -\sum_{i=1}^{n} \log p(y_i|x_i,\theta), \tag{4.3.11}$$

since $p_{\text{emp}}(y_i|x_i) = 1$.

Using the the Shannon-Fano code, which assigns codeword lengths as the inverse of the probabilities, $l_i^{SF} = \lceil -\log p_i \rceil$ is near optimal (cf. Sec. 4.3.2). Thus, if the data x^n, y^n really are generated by f_θ independently, the cross-entropy (4.3.10) equals, up to a single bit, the optimal codelenght.

$$\begin{aligned} l^{SF}(y^n|x^n) &:= \lceil -\log p(y^n|x^n,\theta) \rceil \\ &= \lceil -\log p(y_1|x_1,\theta) \cdots p(y_n|x_n,\theta) \rceil \\ &= \left\lceil -\sum_{i=1}^{n} \log p(y_i|x_i,\theta) \right\rceil \\ &= \lceil H(Y_{\text{emp}}, f_\theta(X_{\text{emp}})) \rceil \end{aligned}$$

4.3.8 The need to replace stochastic complexity minimization to select DNN classifiers

As we saw in Sec. 4.3.5, stochastic complexity can be written in eq. 4.3.9 as the sum of two terms: the first term is the maximum log-likelihood of the training data for models in the parametric family. This can be seen as a measure of noise in the data, as seen by the model family. The second term is what we called the model cost and, as we saw above, can be interpreted as a measure of complexity of the model family.

This measure of complexity does not depend on a particular choice of encoding, which resolves the ambiguity in 2-part MDL. It does so by defining complexity in terms of the behavior of the model family with respect to *data*.

Unfortunately, as we shall now argue, this measure of complexity is not appropriate for a neural network classification setting in a sufficiently overparameterized regime. It is hence not an appropriate tool for model selection in this context. For three reasons: (i) the model cost term is infinite, as we are summing over all possible data, a difficulty that is common in NML for MDL [GR19]. Even if this difficulty is addressed, for example by considering only model families with large but finite support or by introducing a luckiness function [GR19] (essentially a prior on the data), two

[11] Because $H(p,q) = \text{KL}(p,q) + H(p)$, their minimizer over q is the same distribution.

difficulties remain: (ii) the noise terms vanishes in the overparameterized regime. For sufficiently large neural network models, a set of parameters exists such that the log-likelihood of the training data is zero. This makes stochastic complexity depend on non-observed data only and, as we shall see in the sequel, in an overly simple way. (iii) For sufficiently differentiable model families, the model cost term is simple far from observations and complex in a small neighborhood of the observations. There, moreover, it depends on whether the directions are relevant for classification (signal) or not (noise).

As we shall seen in the sequel in Sec. 4.4.1, this motivates our introduction of an MDL principle that implicitly defines model complexity in terms of signal and noise in classification data.

Noise term is zero in the overparameterized regime Feedforward neural networks are universal approximators [HSW89; Cyb89], as they can approximate arbitrary continuous functions on compact sets to any desired degree of accuracy, given a sufficient number of parameters. In a number of cases of interest for neural network classifiers, we indeed work in an overparameterized regime, where the number of parameters of the model far exceeds the number of training data. In that case, one is typically interested in classification accuracy. As it was shown in [Zha+17], in this situation neural networks can memorize labels of arbitrary data with remarkable ease and reach perfect accuracy. Moreover, if the network is sufficiently overparameterized, the training loss continues to decrease during training[12], eventually improving test performance, a phenomenon know as *deep double descent* [Nak+21b] in the literature. Indeed, there is work showing that specifically preventing the loss to decrease improves performance [Ish+20].

In the sequel we shall assume categorical cross entropy loss as discussed in Sec. 4.3.7, and i.i.d. observations. In this case then, loss coincides with the noise term $-\log p(y_o^n|x_o^n, \hat{\theta}(z_o^n))$ in eq. 4.3.9:

$$\begin{aligned} 0 \leq -\log p(y_o^n|x_o^n, \hat{\theta}(z_o^n)) &= -\sum_{i=1}^{n} \max_{p\in\mathcal{M}} \log p(y_o^i|x_o^n) \\ &= \min_{p\in\mathcal{M}} -\sum_{i=1}^{n} \log p(y_o^i|x_o^n) \\ &\leq -\sum_{i=1}^{n} \log p(y_o^i|x_o^n) \\ &= H(Y_{\text{emp}}, f_\theta(X_{\text{emp}})) \to 0 \end{aligned}$$

Where in the first and second steps we used independence of observations, in the step before last we used eq. 4.3.10, and the last step is by assumption in the overparameterized regime[13]. It follows that for sufficiently complex neural network classifiers, without explicitely imposing measures to prevent the loss to reach zero, the first term in stochastic complexity is zero. This implies that for classifier model families in the overparameterized regime, MDL simply focuses on the behavior of

[12]Note that the second task is more difficult than the first, and requires a "universal approximator" class of overparameterization, rather than a "universal classifier" class of overparameterization.

[13]And we slightly abused the notation introduced earlier in this chapter in order to keep the subscript "o" for "observed"; whereas x^n denotes the set of n observed inputs $x_1, \ldots, x_n$ in the first line, in subsequent lines x^i denotes the i-th observed input.

the model family for unobserved data:

$$\begin{aligned}
\mathrm{SC}(y_o^n|x_o^n, \mathcal{M}^i) &= \overbrace{-\log p(y_o^n|x_o^n, \hat{\theta}(z_o^n))}^{\substack{=0 \text{ in the} \\ \text{overparameterized regime}}} + \mathbf{COMP}_n(\mathcal{M}^i) \\
&= \mathbf{COMP}_n(\mathcal{M}^i) \\
&= \log \left(\sum_{z^n \in \mathcal{Z}^n} p(y^n|x^n, \hat{\theta}(z^n)) \right) \\
&= \log \left(\sum_{z^n = z_o^n} 1 + \sum_{z^n \neq z_o^n \in \mathcal{Z}^n} p(y^n|x^n, \hat{\theta}(z^n)) \right) \\
&= \log \left(1 + \sum_{z^n \neq z_o^n \in \mathcal{Z}^n} p(y^n|x^n, \hat{\theta}(z^n)) \right)
\end{aligned}$$

Hence, given a set of neural network classifier families $\left\{\mathcal{M}^i\right\}_{i=1\ldots n}$ in the overparameterized regime, the MDL principle, which tells us to pick the model family minimizing stochastic complexity, simply prescribes picking the model family minimizing model cost for unobserved data:

$$\mathcal{M}^j \text{ is best explanation of } y_o^n|x_o^n \text{ within model families } \left\{\mathcal{M}^i\right\}_{i=1\ldots n} \Leftrightarrow$$

$$j = \arg\min_i \left\{\mathrm{SC}(y_o^n|x_o^n, \mathcal{M}^i)\right\} \overbrace{\Leftrightarrow}^{\substack{\mathcal{M}^i \text{ are NN in the} \\ \text{overparameterized regime}}} j = \arg\min_i \left\{\mathbf{COMP}_{z^n \neq z_o^n}(\mathcal{M}^i)\right\}$$

where we wrote $\mathbf{COMP}_{z^n \neq z_o^n}(\mathcal{M}^i) := \log \left(\sum_{z^n \neq z_o^n \in \mathcal{Z}^n} p(y^n|x^n, \hat{\theta}(z^n)) \right)$ and ignored the constant inside the logarithm in minimization.

By a simple convergence argument, the maximum likelihood of most unobserved data should be close to zero, even in the infinitely countable data case. Specifically, consider model families $\mathcal{M}^1, \mathcal{M}^1$ containing the empirical distribution of z_o^n. Then MDL prescribes selecting the M^i for which the distributions it contains assign as little likelihood as possible to unobserved data. In particular, a model family containing only the empirical distribution $\mathcal{M} = \{\delta(z^n - z_o^n)\}$ would always be selected. And importantly, regardless of the model family that it selects, MDL assigns as best explanation of the data (the $p \in \mathcal{M}^*$ minimizing stochastic complexity within the model family minimizing stochastic complexity) the empirical distribution!

In the overparameterized regime, MDL thus prescribes, to explain data z_o^n, picking the model family that contains distributions that are point-wise closest in likelihood to the empirical distribution. And within that family, it prescribes picking the empirical distribution as a model for z_o^n. This is clearly not very useful.

Finally, we should add that the argument carries over to the case where the maximum likelihood of the data amongst the model families is the same but not one: MDL would select model families assigning the least maximum likelihood to unobserved data. The actual distribution being selected, however, would not be the empirical distribution.

Differentiable model families There is a saving grace in the case of *differentiable* model families (such as neural networks with common activation functions). In this case, Taylor's theorem allows us to express the behavior of the network for unobserved data in the neighborhood of observations in terms of its behavior at the observed points.

The following qualitative discussion ignores matters of convergence and glosses over other important details, but it serve as a guide for intuition, and to motivate our novel MDL principle.

To make matters as precise as possible, we focus on *neural network model families*, consisting of neural network classifiers trained using data z_t^n by gradient descent to minimize the categorical cross-entropy 4.3.10 loss. Different families correspond to different architectures, initialization methods, gradient descent flavors, etc.

Hence, by $\hat{\theta}(z^n)$ we mean "the set of parameters that, for neural networks trained using data z_t^n by gradient descent to minimize cross-entropy loss", assign maximum likelihood to data $y^n|x^n$"[14].

In addition, to simplify notation, we omit the n superscript in x^n, y^n, z^n and the "hat" in θ. We thus denote $z = (x, y)$, and $p(y|x, \theta(x, y))$ is the maximum likelihood that any model within $\mathcal{M}$ can assign to labels y given inputs x. In particular, in the overparameterized regime, we shall assume that $p(y_t|x_t, \theta(x_t, y_t)) = 1$ as seen in the previous section. Finally, motivated by the discussion in Sec. 4.3.5, we shall assume that $\mathcal{Z}$ is compact and that $\forall z \in \mathcal{Z} \quad \exists \theta : p(y|x, \theta(z)) > 0$. That is, we assume that no observation is impossible for all members of $\mathcal{M}$.

With these assumptions, minimizing stochastic complexity is, as seen above, minimizing the maximum log-likelihood of unobserved data

$$\min \log \left(\sum_{z \neq z_t \in \mathcal{Z}} p(y|x, \theta(z)) \right)$$

If we expand each of these likelihoods around the observations, setting $z = z_t + \delta z$, and assuming for ease of exposition that the number of labels is sufficiently large that taking the derivative with respect to y is a sensible thing to do, we get

$$\begin{aligned}
\sum_{z \neq z_t \in \mathcal{Z}} p(y|x, \theta(z)) &= \sum_{\delta z \neq 0 \in \mathcal{Z}} p(y_t|x_t, \theta(z_t)) + \left(\frac{dp}{dz} \right)^\top \delta z + O(\delta z^2) \\
&= \sum_{\delta z \neq 0 \in \mathcal{Z}} p(y_t|x_t, \theta(z_t)) + \left(\frac{\partial p}{\partial x} + \frac{\partial p}{\partial \theta} \frac{\partial \theta}{\partial x} \right)^\top \delta x + \\
&+ \left(\frac{\partial p}{\partial y} + \frac{\partial p}{\partial \theta} \frac{\partial \theta}{\partial y} \right)^\top \delta y + O(\delta z^2) \\
&= \sum_{\delta z \neq 0 \in \mathcal{Z}} 1 + \left(\frac{\partial p}{\partial x} + \frac{\partial p}{\partial \theta} \frac{\partial \theta}{\partial x} \right)^\top \delta x + \left(\frac{\partial p}{\partial y} + \frac{\partial p}{\partial \theta} \frac{\partial \theta}{\partial y} \right)^\top \delta y + O(\delta z^2)
\end{aligned}$$

where we used $p(y_t|x_t, \theta(z_t)) = 1$ in the overparameterized regime, and $\frac{\partial p}{\partial x}$ denotes the gradient of p with respect to the input x, calculated at the training data x_t, y_t, and similarly for the other terms.

Aiming at understanding the behavior of this expression, we divide our analysis in two parts: (i) $\|\delta z\|$ small and (ii) $\|\delta z\|$ large.

[14] Note the maximization within the set of all networks trained using data z_t^n!

(i) for δz small, the first order terms dominate. For the sum to converge, we must thus have $(dp/dz)^\top \delta z \approx -1$ for almost all δz: that is, in the neighborhood of the observations, *most* directions are those of steepest decline of likelihood. Now if the distance between z and z_t is of the order of $\|\delta z\|^2$, then in the direction $h := z - z_t$, the first order term in the expansion is zero. If the labels don't change ($\delta y = 0$), in input space, the direction δx does not impact classification. As defined by the classification problem, it can be considered as *noise*. By setting the first term in the Taylor expansion to zero, we see that noise directions are orthogonal to directions of maximum change (a decrease) in likelihood in terms of input, and hence have little impact on likelihood as well:

$$\left(\frac{dp}{dx}\right)^\top \delta x = 0$$

If on the other hand a direction δx does impact classification ($\delta y \neq 0$) — and can thus be considered as *signal* — changes in input will have an impact on likelihood as well. In this case, the change (decrease) in likelihood is determined by the equation

$$\left(\frac{dp}{dx}\right)^\top \delta x + \left(\frac{dp}{dy}\right)^\top \delta y = 0,$$

where some additional condition, on the norm of the derivatives for example, would have to be imposed to obtain a single solution for each δz.

(ii) Let us now look at the case where δz is large. In this case the likelihood function is no longer dominated by the first-order term. To minimize stochastic complexity in this case we must assign a likelihood as close to zero as possible for *all* δz, which, as we saw above, is achieved quite rapidly outside a small neighborhood of the observations.

In conclusion, in the overparameterized regime, in the case of differentiable model families such as neural network classifiers, MDL as a principle of model selection focuses on the behavior of the model family in a small neighborhood of the observations. In that region, likelihood should decrease in almost every direction (noise) as rapidly as possible, except in the directions that impact classification (signal). The "velocity" of the descent is determined by the behavior of the particular model family with respect to change in inputs and labels in the neighborhood of the training data, i.e. the derivative terms in the Taylor expansion above. The greater the variety (as measured by the size of the derivatives) of networks that can be obtained with the same training data, the steeper the descent.

The discussion above motivates our MDL principle, adapted to neural network classifiers in the case where the number of parameters far exceeds the training data. Instead of measuring complexity with respect to all data, we focus on the behavior of the model family with respect to signal and noise, as defined by the classification task.

4.4 Learning with a novel MDL principle

We now provide an MDL principle that eliminates the need for defining the model encoding, as in two-step MDL or a universal coding such as one-step MDL [GR19]. Instead, we utilize the signal and the noise in the training data to implicitly define the encoding: increase the description length of the noise, and decrease the description length of the signal.

We then establish a lower bound of this maximization objective in terms of the minimal description lengths of signal and noise(cf. 4.2). We further simplify the problem by expressing it locally,

which enables us to provide an interpretation in terms of sensitivities to the signal and noise. Finally, we combine these local problems to express a global MDL objective in terms of the spectra of the local Jacobians, and that the spectral distribution of models that maximize MDL is either power law or lognormal.

4.4.1 MDL objective

The MDL paradigm quantifies learning based on the ability to compress: if $f(X+\Delta)$ contains information about X it can compress it and conversely, if it does *not* contain information about the Δ, it cannot be used to compress it. This formulation implicitly defines the complexity of the model f in terms of unknown X and Δ present in training data. It is therefore applicable in a classification context, where these are defined with respect to a *task*. Formally:

Definition 27 (MDL principle)**.** *Let $\tilde{X} = X + \Delta$ be noisy data, comprised of unknown signal X and a noise Δ parts in the sense of 26, and a model f_θ trained on $\tilde{X}$ according to some (e.g. classification) objective. Let $\mathcal{L}(X|f(\tilde{X}) = y)$ and $\mathcal{L}(\Delta|f(\tilde{X}) = y)$ be, respectively, the expected description length of X and Δ given knowledge $f_\theta(\tilde{X}) = y$. Then with $\gamma > 0$ a hyperparameter, f_θ follows the MDL principle if it maximizes*

$$\max_\theta \left\{ \int p_{f_\theta(\tilde{X})}(y)\mathcal{L}(\Delta|f_\theta(\tilde{X}) = y)dy - \gamma \int p_{f_\theta(\tilde{X})}(y)\mathcal{L}(X|f_\theta(\tilde{X}) = y)dy \right\} \tag{4.4.1}$$

The idea is to minimize the mean $\mathcal{L}(X|f(\tilde{X}) = y)$ and maximize $\mathcal{L}(\Delta|f(\tilde{X}) = y)$ seen as functions of y[15], with γ controlling the relative strength of these objectives.

A lower bound in terms of minimal description length Using Theorem 4.2 we can express the length of the description of noise knowing $f_\theta(\tilde{X}) = y$ as a multiple $\alpha(y) \geq 1$ of the length of the minimum length description for each y:

$$\begin{aligned}
\int p_{f_\theta(\tilde{X})}(y)\mathcal{L}(\Delta|f_\theta(\tilde{X}) = y)dy &= \int p_{f_\theta(\tilde{X})}(y)\alpha(y)H(\Delta|f_\theta(\tilde{X}) = y)dy \\
&\geq \left(\inf_y \alpha(y)\right) \int p_{f_\theta(\tilde{X})}(y)H(\Delta|f_\theta(\tilde{X}) = y)dy \\
&= \left(\inf_y \alpha(y)\right) H(\Delta|f_\theta(\tilde{X}))
\end{aligned}$$

Proceeding similarly for the signal term we obtain

$$\int p_{f_\theta(\tilde{X})}(y)\mathcal{L}(X|f_\theta(\tilde{X}) = y)dy \leq \left(\sup_y \beta(y)\right) H(X|f_\theta(\tilde{X}))$$

Denoting $\inf_y \alpha(y) := \alpha$ and $\sup_y \beta(y) := \beta$ the minimum and maximum expected description lengths of codes of noise and of signal, respectively, knowing $f_\theta(\tilde{X}) = y$, we combine the two desiderata and maximize a lower bound of 4.4.1:

$$\max_\theta \left\{ \alpha H(\Delta|f_\theta(\tilde{X})) - \gamma\beta H(X|f_\theta(\tilde{X})) \right\}$$

[15]For classification, we work on an intermediate representation, which explains the use of integrals in calculating the expectation.

Since $H(\Delta|f_\theta(\tilde{X})) = H(\Delta, f_\theta(\tilde{X})) - H(f_\theta(\tilde{X}))$ and similarly for the second term,

$$H(\Delta|f_\theta(\tilde{X})) - \gamma\beta H(X|f_\theta(\tilde{X})) = \alpha H(f_\theta(\tilde{X})|\Delta) - \gamma\beta H(f_\theta(\tilde{X})|X) + \alpha H(\Delta) - \gamma\beta H(X) + (\beta\gamma - \alpha)H(f_\theta(\tilde{(}X)))$$

Ignoring terms independent of θ, since $\alpha > 0$, we obtain a lower bound of 4.4.1:

Proposition 4.2 (MDL objective lower bound)**.** *Given noisy data $\tilde{X} = X + \Delta$ comprised of a signal X and a noise Δ parts, a model f_θ trained on $\tilde{X}$ according to MDL, $\lambda := \gamma\frac{\beta}{\alpha}$, the following is a lower bound of the the MDL objective:*

$$\max_\theta \Big\{ H(f_\theta(\tilde{X})|\Delta) - \lambda H(f_\theta(\tilde{X})|X) + (\lambda - 1)H(f_\theta(\tilde{X})) \Big\} \tag{4.4.2}$$

In this lower bound, λ has the role of γ modulated by the ratio between the worst case expected signal description length knowing the model output and the best case description length of the noise knowing the model output in units of entropy. Note that to minimize the description length of the noisy data $H(f_\theta(\tilde{X}))$ we must have $\lambda - 1 < 0$ and hence objective 4.4.2 is MDL in expectation with a constraint on the conditional entropies. Since $\lambda < 1 \Rightarrow \alpha > \gamma\beta$ the implications depend on the model class $\{f_\theta\}$: if for the given model class Δ is more difficult to compress than X, then $\alpha > \beta$ and so $\gamma < 1$. This corresponds to, in 4.4.2, focusing relatively more on ignoring the noise. Conversely, if $\{f_\theta\}$ is such that X is mode difficult to compress, then $\gamma > 1$ and we focus relatively more on learning the signal.

As an example, consider F-MNIST data noised up by the addition of MNIST data, and a model class consisting of an intermediate representation of a classifier trained on MNIST, which we hold fixed, to which we add a final trainable classification layer. We expect that members of this model class be better at representing noise (MNIST), hence $\gamma > 1$. Learning in this case corresponds mostly to finding the element of the model class that is better at representing the signal. Conversely, if the representation is learned on MNIST $\gamma < 1$ and we expect learning to correspond to finding the model that is best able to ignore F-MNIST data.

4.4.2 Local formulation

We now simplify the problem in 4.4.2 by expressing it *locally* and then ultimately in terms of the spectrum of the point Jacobian matrix $\nabla f_\theta|_{x_k}$.

Local objective

We begin by showing that given a sufficiently smooth function f, points $x_1, \ldots, x_N$ and an error budget E, a set of radii can be chosen such that the maximum linear approximation error does not exceed it, and these radii are inversely proportional to the largest principal singular value of the point Hessian matrices, a result that is a direct consequence of Taylor's theorem. We also show, conversely, that for a compact domain, a set of radii can be chosen such that every point is inside one of the neighborhoods of the $x_1, \ldots, x_N$ that minimizes the total approximation error.

Intuitively, since the Hessian matrix at a point controls the curvature, the curvature along the maximum curvature direction controls how far we are able to go away from the point while not changing the Jacobian too much.

Lemma 4.1. *Let $f : A \subseteq \mathbb{R}^n \to B \subseteq \mathbb{R}^m$ be analytical, with A compact and $x_1, \dots, x_N \subseteq A$. Then given $E^k > 0$, a set of balls $\{V_k\}_{k=1\cdots N}$ centered at x_k and with radius r_k can be chosen such that the norm of the approximation error $f(x_k + r_k) - f(x_k)$ is upper bounded by*

$$\forall_{k=1\cdots N},\ \sup_{\substack{w\in V_k \\ i=1\cdots m}} \frac{1}{2}\sigma_1(\nabla^2 f^i|_w) \cdot \|r_k\|^2 = E^k$$

where $\sigma_1(\nabla^2 f^i|_w)$ is the first singular value of the Hessian matrix of the component f^i calculated at $w \in V_k$.

Conversely, a set of radii can be chosen such that every point is inside one of the neighborhoods of the $x_1, \dots, x_N$ that minimizes the total approximation error.

Proof. For each component f^i of f, Taylor's theorem states that the approximation error of $f^i(x_k + r_k) - f^i(x_k) \approx \left(\nabla f^i|_{x_k}\right) r_k$, along a radius r_k, in Lagrangean form, is $\frac{1}{2} {r_k}^\top \left(\nabla^2 f^i|_w\right) r_k$, where w is a point between $x_k, x_k + r_k$. The approximation error is thus

$$\begin{aligned}\frac{1}{2} {r_k}^\top \left(\nabla^2 f^i|_w\right) r_k &= \frac{1}{2} \frac{{r_k}^\top \left(\nabla^2 f^i|_w\right) r_k}{{r_k}^\top r_k} \cdot \|r_k\|^2 \\ &\leq \frac{1}{2}\sigma_1(\nabla^2 f^i|_w) \cdot \|r_k\|^2 \\ &\sup_{\substack{w\in V_k \\ i=1\cdots m}} \frac{1}{2}\sigma_1(\nabla^2 f^i|_w) \cdot \|r_k\|^2\end{aligned}$$

, where we used the definition of the first singular value in terms of the Rayleigh quotient to establish this result on the sup norm. A result that holds for other norms follows from convexity of the norms and the bound on each of the components of the vector of the Hessian matrices.

To see the converse, consider that given a compact set and a point, there is always a ball that contains it. Hence so would a union of such balls. Since each of the radii sets an upper bound for the local approximation error, with $\sup_{\substack{w\in V_k \\ i=1\cdots m}} \frac{1}{2}\sigma_1(\nabla^2 f^i|_w) := \sigma_1^k$, we can write the total error as

$$\min_{r_k} \sum_{k=1}^{N} \sigma_1^k \cdot r_k^2$$

with the constraint that $A \subseteq V_1 \cup \cdots \cup V_N$. □

Proposition 4.3 (Local MDL objective)**.** *In the conditions of Lemma 4.1 and notation above, locally in V_k the MDL objective 4.4.2*

$$\max_f \left\{ H(f(\tilde{X})|\Delta) - \lambda H(f(\tilde{X})|X) + (\lambda - 1)H(f(\tilde{X})) \right\}$$

can be expressed approximately as

$$\max_{J_k} \lambda H(J_k \delta X_k) - H(J_k \Delta_k) \tag{4.4.3}$$

where $\delta X_k, \Delta_k$ denote the signal and the noise in V_k with respect to its center, and the approximation error is controlled by Prop. 4.1.

Proof. Let $f : A \subseteq \mathbb{R}^n \to B \subseteq \mathbb{R}^m$ be analytical, A compact and $\mathcal{D} = x_1, \ldots, x_N \subseteq A$ and $\{V_k\}_{k=1\cdots N}$ a set of balls centered at x_k and with radius r_k such that $A \subseteq V_1 \cup \cdots \cup V_N$, chosen such that the Jacobian matrix of f is constant in each V_k in the sense of Lemma 4.1. Then to first order in $\delta x_k, \delta$, every $x \in A$ can be expressed in terms of the "center" of the V_k that contains it $k(x) = k \in \{1, \cdots, N\} : x \in V_k$. Writing for simplicity $x_{k(x)} := x_k$ we have, for all $x \in A$

$$\begin{aligned} f(\tilde{x}) &= f(x_k + \delta x_k + \delta_k) \\ &\approx f(x_k) + \nabla f|_{x_k} \delta x_k + \nabla f|_{x_k} \delta_k \\ &:= f(x_k) + J_k \delta x_k + J_k \delta_k \end{aligned}$$

with the approximation error controlled by the principal singular value of the Hessian.

Noting that the choice of V_k determines $f(\tilde{x}_k)$, we can apply this approximation to 4.4.2 to obtain a local version, in V_k, of the expression to maximize :

$$H(J_k \delta X_k + J_k \Delta_k | \Delta_k) - \lambda H(J_k \delta X_k + J_k \Delta_k | \delta X_k)$$

Assuming local independence of signal and noise implies locally that $H(J_k \delta X_k | \delta X_k) = 0, H(J_k \Delta_k | \Delta_k) = 0$ and $H(J_k \delta X_k | \Delta_k) = H(J_k \delta X_k), H(J_k \Delta_k | \delta X_k) = H(J_k \Delta_k)$? Hence:

$$\begin{aligned} &H(J_k \delta X_k + J_k \Delta_k | \Delta_k) - \lambda H(J_k \delta X_k + J_k \Delta_k | \delta X_k) + (\lambda - 1) H(J_k \delta X_k + J_k \Delta_k) = \\ &H(J_k \delta X_k) - \lambda H(J_k \Delta_k) + (\lambda - 1) \left(H(J_k \delta X_k) + H(J_k \Delta_k) \right) = \\ &\lambda H(J_k \delta X_k) - H(J_k \Delta_k) \end{aligned}$$

Finally, noting that maximizing f locally in V_k, at this order of approximation, amounts to maximize over the its local Jacobian

$$\max_f \lambda H(J_k \delta X_k) - H(J_k \Delta_k) \overset{V_k, \text{order of approx.}}{\Leftrightarrow} \max_{J_k} \lambda H(J_k \delta X_k) - H(J_k \Delta_k)$$

which proves the claim. □

Interpretation in terms of sensitivity measure in [Aro+18]

In [Aro+18] the authors define sensitivity of a mapping f with respect to noise Δ at x as

$$S = \mathbb{E}_{\delta \sim \Delta} \left[\frac{\|f(x+\delta) - f(x)\|^2}{\|f(x)\|^2} \right]$$

In a region of constant Jacobian J_k, using the arguments in 4.4.2, to first order in δ, we obtain

$$\frac{\|f(x+\delta) - f(x)\|^2}{\|f(x)\|^2} \approx \frac{\|J_k \delta_k\|^2}{\|f(x_k)\|^2}$$

In expectation, up to a scale, this is the variance of $J_k \Delta_k$ which is a measure of its complexity like the entropy above, (for a Gaussian distribution, up to a logarithm and a constant, the two coincide). $H(J_k \Delta_k)$ in prop. 4.3 thus corresponds to sensitivity with respect to noise and, by a similar argument, $H(J_k \delta X_k)$ to sensitivity with respect to signal. Our MDL objective thus selects the model that locally maximizes sensitivity with respect to signal and minimizes sensitivity

with respect to noise. Although similar to [Aro+18], in our formulation sensitivity is logarithmic, direction-dependent (cf.4.4.2), and crucially combines sensitivity to signal and sensitivity to noise.

Finally, since $\lambda < 1$, if $H(J_k\delta X_k) > H(J_k\Delta_k)$ then 4.4.3 is upper bounded by zero, where $\lambda = \frac{H(J_k\Delta_k)}{H(J_k\delta X_k)}$. Maximizing 4.4.3 thus corresponds to getting closer to a model that *locally* produces the same balance between sensitivity to signal and to noise, determined by the *global* parameter λ. This problem cannot always be solved. Consider f a one layer ReLU network of width N; the *local* $\{J_k\}$ are given by deleting a certain number of rows in the pre-ReLU Jacobian, which is the weight matrix of f. Since f can have at most 2^N different $\{J_k\}$, the conjunction of local problems can only be solved if the number of V_k where the balance between sensitivities needs to be adjusted *differently* is smaller that 2^N. The case of deeper networks is similar, each new ReLU layer of width M_i multiplying the number of possible Jacobians by 2^{M_i}.

Local objective: spectral formulation

To provide a spectral version of 4.4.2, we express J_k in terms of its singular value decomposition (SVD), and the signal and noise in terms of local PCA representations. We work in V_k but omit the label k wherever possible for clarity of presentation. Jacobian, signal, and noise refer to the *local* versions.

Proposition 4.4. *(Local objective spectral formulation) Let δX_{pca} and Δ_{pca} be, respectively, the representations of the local signal δX and noise Δ in a neighbourhood of constant Jacobian V_k, and $\delta X^j_{pca}, \Delta^j_{pca}$ its k-th component (with basis vectors ordered by the magnitude of its associated eigenvalue). In the conditions of prop. 4.3, the following is its lower bound:*

$$\max_{\sigma} \left\{ \lambda \left(\max_i \left\{ \log \sigma_i + H(\delta X^i_{pca}) \right\} \right) - \sum_j \left(H(\Delta^j_{pca}) + \log \sigma_j \right) \right\} \tag{4.4.4}$$

Proof. Let $J = U\Sigma V^\top$ be the singular value decomposition of $J \in \mathbb{R}^{n\times m}$. Drawing from the argument in the proof of Proposition 4.3, for each observation $\tilde{x}^{(i)} \in V_k$, we have $\tilde{x}^{(i)} = x_k + \delta x^{(i)}_k + \delta^{(i)}_k$. Each point in V_k can thus be represented by its local signal $\delta x^{(i)}_k$ and noise vectors $\delta^{(i)}_k$.

The set of all noise vectors in V_k induces a PCA basis, composed of the eigenvectors of the noise data matrix $\left[\delta^{(1)}_k \; \delta^{(2)}_k \; \dots \; \delta^{(m)}_k\right]$ and similarly for the signal data.

Thus, the signal δX can be expressed as the transform to local coordinates of δX_{pca}, the signal in local PCA coordinates $\delta X = W^\top_{signal}\delta X_{pca}$, and similarly for noise: $\Delta = W^\top_{noise}\Delta_{pca}$, where W_{signal}, W_{noise} are, respectively, the PCA coordinate transformation for signal and for noise[16]. Noting that U has determinant one everywhere, the transformation it induces does not change the entropy. We thus have:

$$\begin{aligned} \lambda H(J\delta X) - H(J\Delta) &= \lambda H(U\Sigma V W^\top_{signal}\delta X_{pca}) - H(U\Sigma V W^\top_{noise}\Delta_{pca}) \\ &= \lambda H(\Sigma V W^\top_{signal}\delta X_{pca}) - H(\Sigma V W^\top_{noise}\Delta_{pca}) \end{aligned}$$

The $VW^\top$ are contractions measuring the alignment between the singular vectors of the Jacobian and the principal components of the signal (for W_{signal}) and noise (for W_{noise}). We thus maximize the RHS of this expression by:

[16] These matrices are, respectively, the transpose of the eigenvector matrix of the signal, and of noise.

- aligning J with δX and then maximizing the logarithm of the singular values in the non-zero dimensions: if δX is locally low-dimensional, the singular values that get maximized are few.
- aligning J with Δ and then minimizing the logarithm of the singular values in the non-zero dimensions: since Δ tends to be relatively high-dimensional, all singular values of J tend to be minimized.[17]

The overall effect is to maximize a few neighborhood-dependent singular values of J, and minimize all the rest – consistently with the experimental observations in Fig. 4.2. See Fig. 4.1 for a schematic representation.

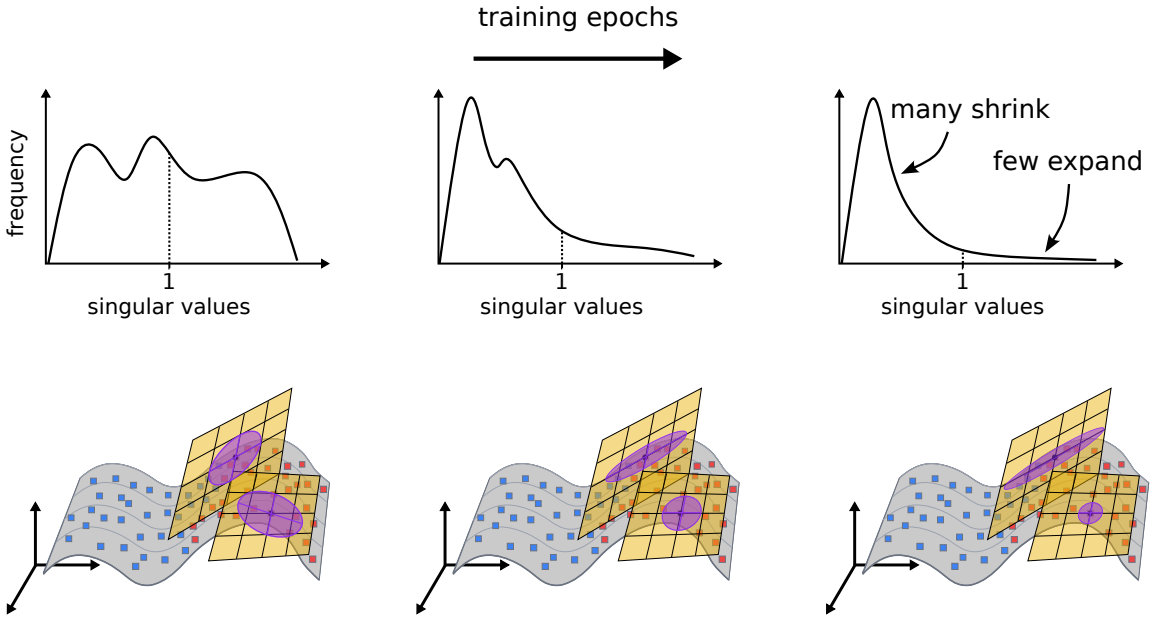


Figure 4.1: Illustration of the evolution of the point Jacobian spectral distribution for a Neural Network classifier at three different epochs – from left to right, (i) initial, (ii) transient, (iii) final. (i, top) Before training, the sensitivity of the network to direction in input space (the wavy plane below) is independent of the direction, as can be seen by the ellipses whose axes' direction/magnitude represent the singular vectors/values of the Jacobian matrix of the network at that point. As can be seen in (i,bottom), the distribution of the spectra of all Jacobian matrices at all training points is, correspondingly, evenly split amongst contractions (singular values smaller than one) and expansions (singular values greater than one) (ii, top) As training evolves, the network becomes increasingly sensitive to directions that are important to classification and less so for directions that are not. This can be seen by the stretching of the ellipses in directions for which the classification changes and shrinking otherwise and, in (ii,bottom), the relative increase of contraction directions. In (iii), we observe the overall final effect on the Jacobians and the spectra.

Since δX and J are unknown, so are the "selected" directions. The full entropy of the local signal is at least as that of its components. Replacing it with the entropy of the singular direction

[17] A similar argument can be found in [Aro+18] in the discussion of noise sensitivity.

i for which the entropy of the transformed signal is maximal, we obtain the stated lower bound of the local objective 4.4.3. □

4.4.3 Combining local objectives to obtain a spectral distribution

We combine local objectives by maximizing their sum over all local patches V_k. This is essentially assuming cross-patch independence. For it to hold, (i) the network should be able to produce sufficiently many local Jacobians as explained in 4.4.2 and (ii) $V_i \cap V_j$ should be small for all i, j. Assumption (i) holds in practice since we work in the overparameterized regime and (ii) holds for ReLU networks. Both assumptions are thus expected to hold as a first approximation, although [HS19] suggests more complex behavior and will be considered in future work.

Recalling that we do not know which singular value gets "selected" and assuming that the signal is locally low-dimensional (which is known as "the manifold hypothesis" [Cay05; FMN16]), which we take for simplicity to mean that $\max_{i_k} H(\delta X_k^{i_k}) \approx H(\delta X_k)$ we obtain, summing over the M patches of rank-N_k Jacobian

$$\sum_{k=1}^{M} \left\{ \lambda \left(\max_{i_k} \left\{ \log \sigma_{i_k} + H(\delta X_k^{i_k}) \right\} \right) - \sum_{j=1}^{N_k} \left(H(\Delta_k^j) + \log \sigma_j \right) \right\}$$

Simplifying and maximizing over the singular values of all the J_k leads to

$$\max_{\sigma} \left\{ \lambda M \mathbb{E}\left[\log \sigma\right] + \lambda H(X) - H(\Delta) - \bar{N} M \mathbb{E}\left[\log \sigma\right] \right\}$$

where expectations of both log singular values and Jacobian rank are over the patches, the latter denoted $\bar{N}$ for readability. As the sum of lower bounds of non-positive quantities is non-positive, its maximum value is zero, where

$$\mathbb{E}\left[\log \sigma\right] = \frac{H(\Delta) - \lambda H(X)}{M(\lambda - \bar{N})} \tag{4.4.5}$$

A note on selecting singular values Aligning the Jacobian and δX implies that the entropy $H(\delta X^{i_k})$ is maximal. Using the manifold hypothesis, we assume that the maximal entropy component accounts for most of the entropy, that is $\max_{i_k} H(\delta X_k^{i_k}) \approx H(\delta X_k)$. As explained in 4.4.2, the maximal entropy component does not necessarily correspond to the maximal singular value: during training, singular spaces corresponding to higher-order singular values will also be "selected". Taking this "selection" as a one-sample estimate of the mean justifies replacing the maximization over i_k in the expression above with $\mathbb{E}[\log \sigma_k] + H(\delta X_k)$. Under the assumption of cross-patch independence, we have $\sum_{i=1}^{M} H(\delta X_k) = H(X)$ and similarly for Δ. The expression below follows from linearity of expectation:

$$\max_{\sigma} \left\{ \lambda M \mathbb{E}\left[\log \sigma\right] + \lambda H(X) - H(\Delta) - \bar{N} M \mathbb{E}\left[\log \sigma\right] \right\}$$

Since each of the terms in the sum above is negative, this expression is non-positive. At the maximum, we obtain 4.4.5

Point Jacobian spectral expectation as a model-dataset measure of complexity For this expectation to be positive the entropy of the noise must be sufficiently smaller than the entropy of the signal, since $\lambda - \bar{N} < 0$ because $0 < \lambda < 1$. If 4.4.1 holds, $\mathbb{E}[\log \sigma]$ thus decreases with the number of patches of constant Jacobian and the mean Jacobian rank. It is thus a measure of model complexity which increases with the weighted difference between the entropy of noise and the entropy of signal, *it depends on the signal and noise*. All things being equal, for the same $\mathbb{E}[\log \sigma]$ models trained with more noise will have smaller M and $\tilde{N}$. Adding noise is a form of regularization. If on the other hand, entropy of noise is greater than the entropy of signal, the reverse effect is produced. On very noisy data (relative to signal!), models trained with more noise need to become more complex.

4.4.4 The MDL spectral distributions

We now show that the predicted distribution that is compatible with 4.4.5 is a power law or, for NN trained with SGD, a lognormal distribution. The true spectral distribution contains information on, e.g. architecture and training process whereas in the maxent formalism [Jay78] we use, the prediction is maximally non-committal: it contains no information on the MDL-trained network beyond its adherence to the MDL principle and the signal-to-noise entropies of the training data.

Incorporating knowledge of the expectation of the log spectrum and SGD The distribution that incorporates knowledge of the expectation of the spectrum4.4.5 *and nothing else* is the maximum entropy distribution for which the constraint on the spectrum 4.4.5 holds [Jay78]. Specifically, the power law distribution

$$p(\sigma) = \frac{\alpha - 1}{\alpha} \left(\frac{\sigma}{b}\right)^{-\alpha},$$

where $\alpha = 1 + \frac{1}{\mathbb{E}[\log \sigma] - \log b}$ and b is a cutoff parameter. Power laws model scale-free phenomena[18], but can emerge when aggregating data over many scales [Zha+15; GC04], as we did in 4.4.3 to obtain eq. 4.4.5. For a ReLU NN trained by SGD, there is also a constraint on the *variance* of $\log \sigma$: the spectrum depends continuously on the network weights (cf. sec. 4.4.2), which are SGD-updated using a *finite* number of steps. The corresponding maxent distribution is the lognormal, which is the Gaussian distribution with given mean and variance in log-scale.

4.5 Experimental results

Our experiments show that spectral distribution matches theoretical predictions in 4.4.4, suggesting that NN are driven by the MDL principle. We study the effect of noise in the point Jacobian spectral distribution of three groups of models of increasing complexity, ReLU MLPs, Alexnet, and Inception trained on MNIST [Den12] and cifar-10 [KH+09], using the experimental setup in [Zha+17]. See Sec. 4.5.1 for details. The section is organized as follows: (i) we describe the experimental setup used in the works presented in this chapter (ii) we present two different types of noise and discuss expected consequences with respect to spectral distribution, and (iii) we present and discuss the experimental results.

[18] Since $p(k\sigma) = a\,(k\sigma)^{\alpha} = ak^{\alpha}\sigma^{\alpha}$. Since the constant is a normalization factor, we must have $p(k\sigma) = p(\sigma)$.

4.5.1 Experimental setup

The experimental setup follows [Zha+17] closely. We investigate two image classification datasets, namely the MNIST dataset [LCB10] and the CIFAR10 dataset [KH+09]. Both datasets are composed of 50,000 training and 10,000 validation images, distributed across 10 different classes. In CIFAR10, each image in the dataset has dimensions of 32x32, with 3 color channels. To scale the pixel values into the range of [0, 1], we normalize them by dividing each value by 255. Additionally, we center crop the images to obtain a size of 28x28, and normalize them by subtracting the mean and dividing the adjusted standard deviation independently for each image, adapting the per_image_whitening function in Tensorflow [Mar+15], as presented in [Zha+17]. The same procedure adapted to one channel, except center cropping which is unnecessary since MNIST images are 28x28, is performed on MNIST.

On both datasets, we use two common deep architectures, which were adapted to smaller image sizes/single-channel images: a simplified Inception model [Sze+17] and Alexnet [KH+09]. As in [Zha+17], the simplified Inception model uses a combination of 1x1 and 3x3 convolution pathways, while the simplified Alexnet is constructed using two (convolution 5x5 $\rightarrow$ max-pool 3x3 $\rightarrow$ local-response-normalization) modules followed by two fully connected layers with 384 and 192 hidden units, respectively. We utilize a 10-unit linear layer for prediction, and to calculate the point Jacobians. All architectures employ the standard rectified linear activation functions (ReLU).

We also study two fully connected multi-layer perceptrons (MLPs): one having a hidden layer with 512 units, the other having three hidden layers of the same size.

For all experiments, we train the models using SGD with a momentum of 0.9, using an initial learning rate of 0.01. We apply a decay factor of 0.95 per epoch to adjust the learning rate, and train the models without weight decay, dropout, or any other explicit regularization techniques.

In all experiments, we calculate the Jacobian at the linear layer, using automatic differentiation with Pytorch's torch.autograd.functional.jacobian method. The point Jacobian spectrum is calculated at all training and test examples using Pytorch's torch.linalg.svdvals, which is a port of Numpy's.

The experimental spectral distributions were split at the first deepest trough. The lognormal fit of each modality, the probabilty plots and line of best fit were calculated using scipy [Vir+20].

4.5.2 Experimental Noise

We study two forms of "natural" noise: *label noise*, used in [Zha+17] and *dataset noise*, which consists in adding a lossy compressed version of a similar dataset.

Label noise We focus on instance-independent symmetric label noise [Son+22], which randomly assigns labels to training and test examples unconditionally on example and training label with probability p. Label noise can be modelled realistically using human annotators [Wei+22], but the former choice is closer to the MDL sense 26. In this setting, the entropy of the introduced noise can be estimated as $p \cdot H(X_0)$, since incorrectly labelled examples become noise with respect to the classification task. This allows us to express the numerator of 4.4.5 for the noised dataset in terms of the entropy and noise of the original dataset as $H(\Delta_p) - \lambda H(X_p) = H(X_0) - \lambda H(X_0) + p(1+\lambda)H(X) > H(X_0) - \lambda H(X_0)$. All things being equal, for NN following MDL, the log Jacobian point spectrum increases with the probability of label noise p.

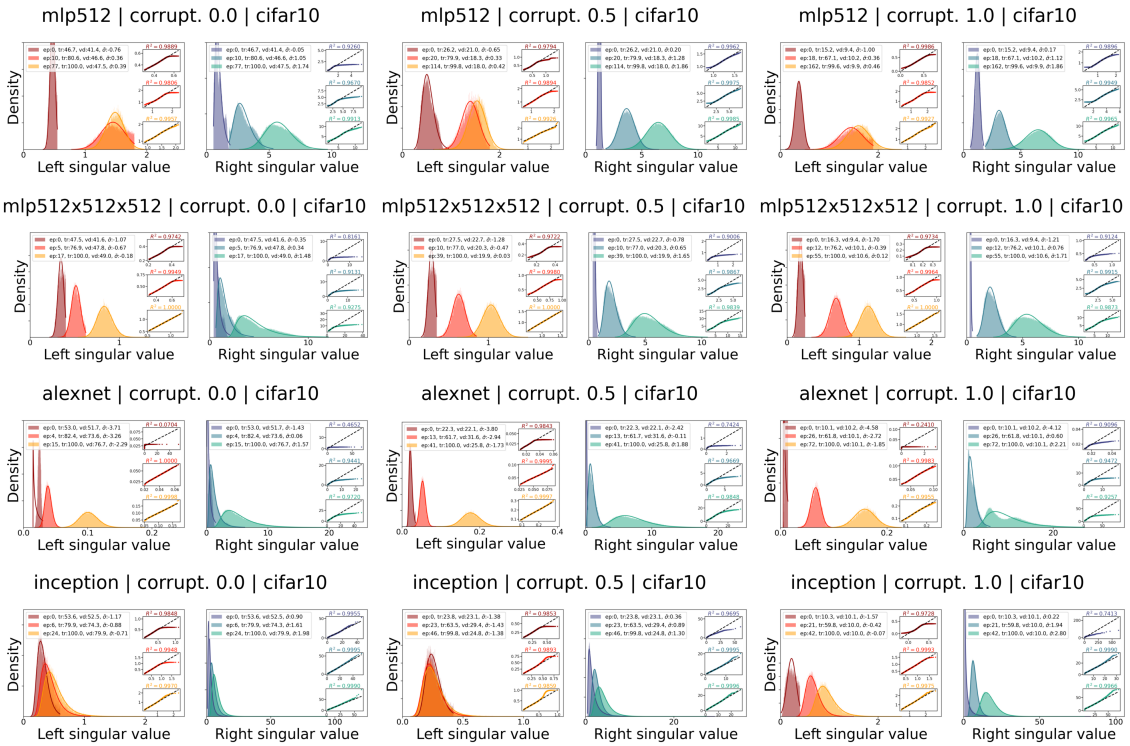


Figure 4.2: Point Jacobian spectral distribution for *model* | ***label noise*** | *cifar-10*, from first epoch to overfit. "Left" and "right" distributions (cf. 4.5.3) are represented separately for each triplet for clarity. The best fit lognormal plot is superimposed on each histogram, with the corresponding probability plot on the right, with the line of best fit (R^2 displayed on top). Legend elements, in order: epoch, training and validation accuracy, and the mean log spectrum.

Dataset noise We add to the original dataset D_0 a *similar* dataset D_{sim} lossy compressed at rate r[19]. Symbolically $D_r = D_0 + rD_{sim}$. We choose D_{sim} commonly used in place of D_0 in ML practice: cifar-100 for cifar-10, and Fashion-MNIST for MNIST. To compress $\tilde{D}$ we reconstruct it using only a certain number of PCA components. This causes less bias in setting r, compared to compressing with e.g. jpeg [PM92] or an autoencoder, in which the architecture introduces an element of arbitrarily, but we lose the ability to set the compression rate at will. Since for the noised dataset $X_r + \Delta_r$ the numerator in 4.4.5 can be written as $H(\Delta_r) - \lambda H(X_r) = H(\Delta_0) - \lambda H(X_0) + r(H(X_{sim}) + H(\Delta_{sim})$. All things being equal, for NN that follow MDL, the average log Jacobian point spectrum decreases with r. Interestingly, assuming the entropies of the similar dataset are approximately the same as that of the original dataset, we obtain $H(\Delta_r) - \lambda H(X_r) = (1+r)H(\Delta_0) - (\lambda - r)H(X_0)$, which corresponds to the same maximization objective with a rescaled

[19] Our notion of dataset noise bears similarities to "bubble noise" in the speech recognition literature [Kob+96], consisting in superimposing independent speech signals. This is a phenomenon that occurs naturally in conversation, and from the point of view of speech recognition, the problem consists in separating *the specific* speech signal that one should be listening to.

$\lambda_r = \frac{\lambda - r}{1+r} < \lambda_0$ corresponding to less sensitivity to signal.

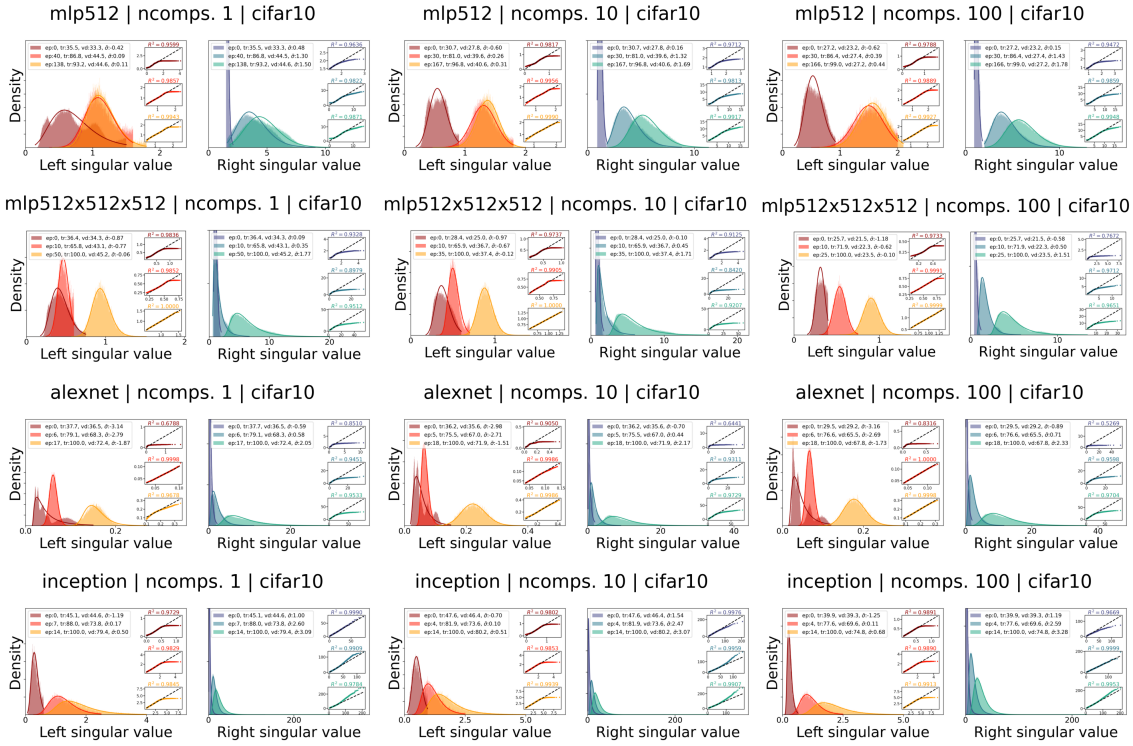


Figure 4.3: Point Jacobian spectral distribution for *model* | ***nbr. pca comp.*** | *cifar-10*, from first epoch to overfit where possible. "Left" and "right" distributions (cf. 4.5.3) are represented separately for each triplet for clarity. The best fit lognormal plot is superimposed on each histogram, with the corresponding probability plot on the right, with the line of best fit (R^2 displayed on top). Legend elements, in order: *epoch*, *training and validation accuracy*, and the *mean log spectrum*.

4.5.3 Discussion

As Figs. 4.2 and 4.3 show, NN trained using SGD are driven by the MDL principle: (i) their spectra is remarkably well-fit by a lognormal distribution, as predicted in 4.4.4, and experimental spectra become globally more lognormal with training epoch (cf. fit overlay on the histograms, and inset probability plots); also, as predicted in the discussion following 4.4.5 (ii) for each model $\mathbb{E}[\log \sigma]$ tends to increase with noise (iii) and with model complexity, which also influences the quality of lognormal fit[20], Inception being the overall best and MLP the overall worst. Remarkably, these observations hold for both label noise and dataset noise. In the early stages of the training process, though, representation-building takes precedence. This can be inferred by observing that experimental distributions are typically bimodal (see Sec. 4.5.3, for figures and discussion), and

[20]The number of training epochs being relatively small, we did not find a power-law behavior.

noting that at the last linear layer of a classification-induced representation, one of the directions should leave the output relatively more unchanged than the others: the direction assigned to the class of the training point (see 4.15 for a visual explanation). Representation building occurs early, as can be seen in Figs. 4.5.3 or in Figs. 4.2 and 4.3, dominating MDL in early epochs. To handle this asymmetry, we divide the spectrum in each of its two modalities (cf. 4.5.1). The statements above apply to each of the two parts of the spectrum, corresponding to the two representations. The observations above hold for MNIST as well, exception being where the initial spectrum is multi-modal (suggesting a great degeneracy of the directions in which the classification prediction does not change — i.e. MNIST is very simple). In this case our splitting method is ineffective, as we would need to split the spectral distribution into each of the several modalities.

Point Jacobian spectrum, full spectra For the figures detailing the full spectrum, i.e. figs. 4.6 through 4.14, note that train and test distributions are the same, since the underlying distributions are the same for this relatively simple dataset. We also note that the overall shape of each distribution does not change significantly with the addition of noise. As discussed in the main text, we note the clear bimodality in all spectral distributions, comparatively higher "lognormality" of Inception, and that the addition of noise increases the mean spectrum.

Finally, note that at the beginning of training, MLP and Alexnet's predictions are very local, since there is a great number of relatively small singular values (high peak), and more so with the addition of noise. This effect is also observed in Inception, but to much less extent. Using fig. 4.15 as illustration, MLP and Alexnet are more *conservative* than Inception, keeping close to the image of the training examples for small perturbations. This is similar to the strategies of generalization that we commonly use (e.g. maxent). Inception, on the other hand, which generalizes better, while not being conservative at all, which suggests that it does so by focusing on the signal.

Point Jacobian spectrum MNIST Figures 4.4 and 4.5 parallel those in the main text for the MNIST dataset.

Dataset noise leading to bimodality Figure 4.15 provides a graphical illustration of the representation building leading to bimodality.

The example is for a classifier $f : \mathbb{R}^3 \to \mathbb{R}^2$ and an autoencoder g for visualization, but the overall idea extends to higher dimensions.

4.6 Conclusion and future work

In this work, we propose an MDL principle that implicitly defines model complexity in terms of signal and noise: *choose the model whose representation of the data can be used to compress the signal, but not the noise.* We show that models driven by this principle locally maximize sensitivity to the signal and minimize the sensitivity to noise, and predict that the point Jacobian spectrum of NN trained by gradient descent follow either a power law or a lognormal distribution. We provide experimental evidence supporting this prediction, hinting that neural networks trained by gradient descent are driven by the MDL principle.

As for future work we plan, aiming at a generalization bound, to extend the connection established in 4.4.2, by making the MDL objective layer wise as in [Aro+18]. Another possible extension

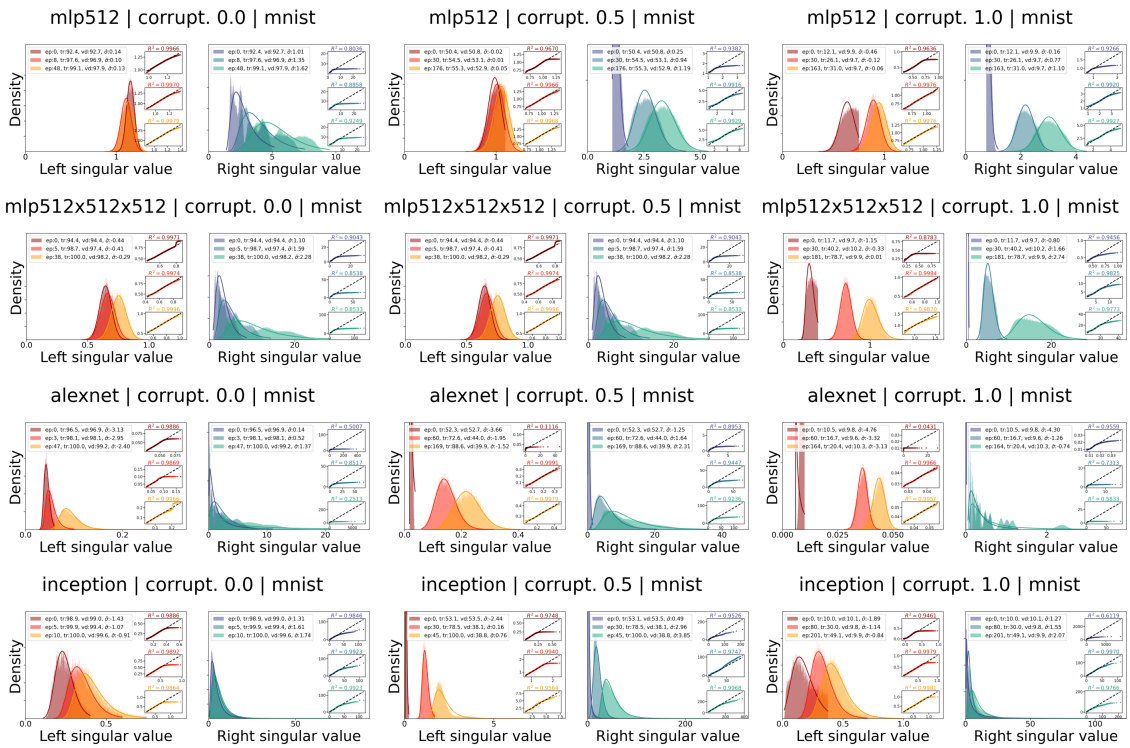


Figure 4.4: Point Jacobian spectral distribution for *model* | ***label noise*** | *MNIST*, from first epoch to overfit. "Left" and "right" distributions (cf. 4.5.3) are represented separately for each triplet for clarity. The best fit lognormal plot is superimposed on each histogram, with the corresponding probability plot on the right, with the line of best fit (R^2 displayed on top). Legend elements, in order: epoch, training and validation accuracy, and the mean log spectrum.

is to use our findings to explain the power law behavior of the spectra of the layer weight matrices and connection to generalization gap found in [MM18; MM20], by noting that each point Jacobian of ReLU networks is a sub-matrix of the product of the network weight matrices, which can be expressed in terms of the singular values of the point Jacobian submatrix via an interlacing inequality [Tho72].

4.6.1 Addendum: deriving the MDL local approximation, alternative derivation

Within each local patch the Jacobian J has a singular vector decomposition $J = U\Sigma V^\top$. Assume without loss of generality that $J \in \mathbb{R}^{n\times m}$. Then $U \in \mathbb{R}^{m\times m}$ is orthogonal, $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix and $V \in \mathbb{R}^{n\times n}$ is orthogonal as well. The entropy $H(V^\top \delta X) = H(\delta X)$ since V^T has determinant one everywhere. In the same way, $H(J\delta X) = H(U^\top J\delta X) = H(\Sigma V^\top \delta X)$. In the case of an embedding, the matrix Σ has a number of zero components along its diagonal, which will, upon multiplying by $V^\top \delta X$ produce a vector that has a number of zero components

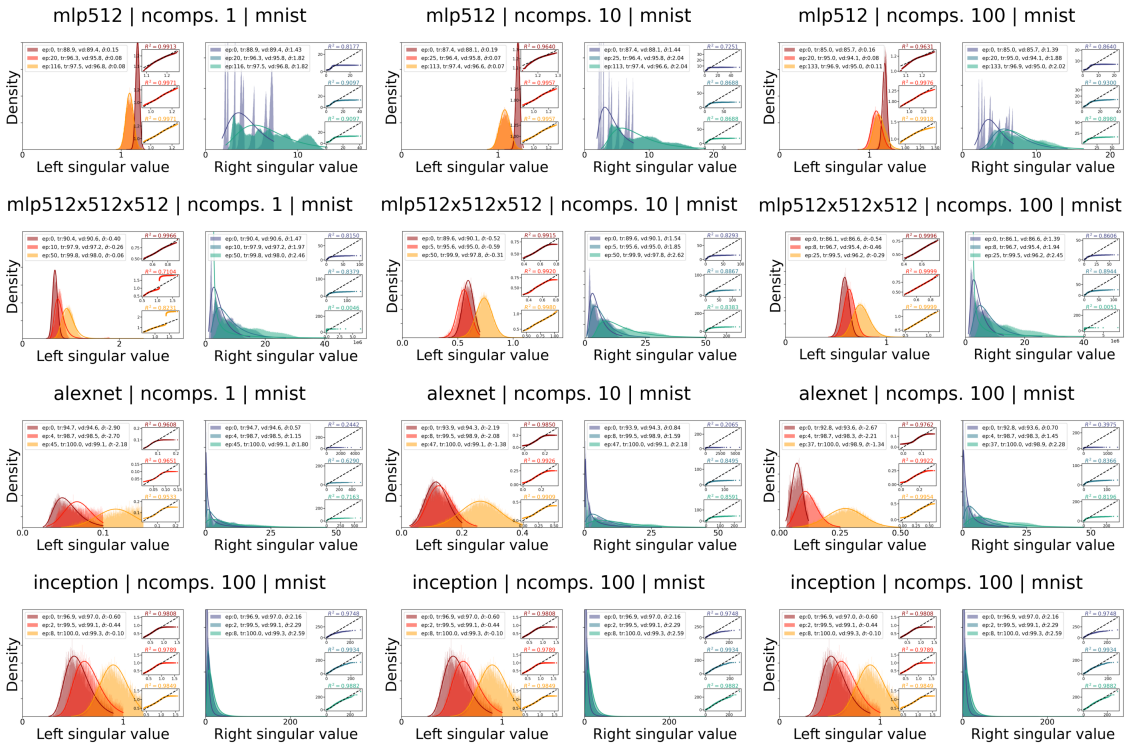


Figure 4.5: Point Jacobian spectral distribution for *model* | ***nbr. pca comp.*** | *MNIST*, from first epoch to overfit. "Left" and "right" distributions (cf. 4.5.3) are represented separately for each triplet for clarity. The best fit lognormal plot is superimposed on each histogram, with the corresponding probability plot on the right, with the line of best fit (R^2 displayed on top). Legend elements, in order: epoch, training and validation accuracy, and the mean log spectrum.

independently of the other components. Hence, the entropy of the zero part and the nonzero part is the same as the entropy of the nonzero part. So the entropy above is just the entropy of the non-zero components after acting on data with $V^\top$. Explicitly, $H(\sigma_1 v_1^\top \delta X, \ldots, \sigma_k v_k^\top \delta X, 0, \ldots, 0) = H(\sigma_1 v_1^\top \delta X, \ldots, \sigma_k v_k^\top \delta X)$. Plugging into our objective we obtain

$$\begin{aligned}\max_J \lambda H(J\delta X) - H(J\Delta) &= \max_\sigma \lambda H(\sigma_1 v_1^\top \delta X, \ldots, \sigma_k v_k^\top \delta X) - H(\sigma_1 v_1^\top \Delta, \ldots, \sigma_k v_k^\top \Delta) \\ &\geq \max_\sigma \left\{ \max_i \lambda H(\sigma_i v_i^\top \delta X) - H(\sigma_1 v_1^\top \Delta, \ldots, \sigma_k v_k^\top \Delta) \right\}\end{aligned}$$

Noting that

$$\begin{aligned}H(\sigma_1 v_1^\top \Delta, \ldots, \sigma_k v_k^\top \Delta) &\leq \sum_i H(\sigma_i v_i^\top \Delta) \\ &= \sum_i H(\Delta_i) + \log \sigma_i\end{aligned}$$

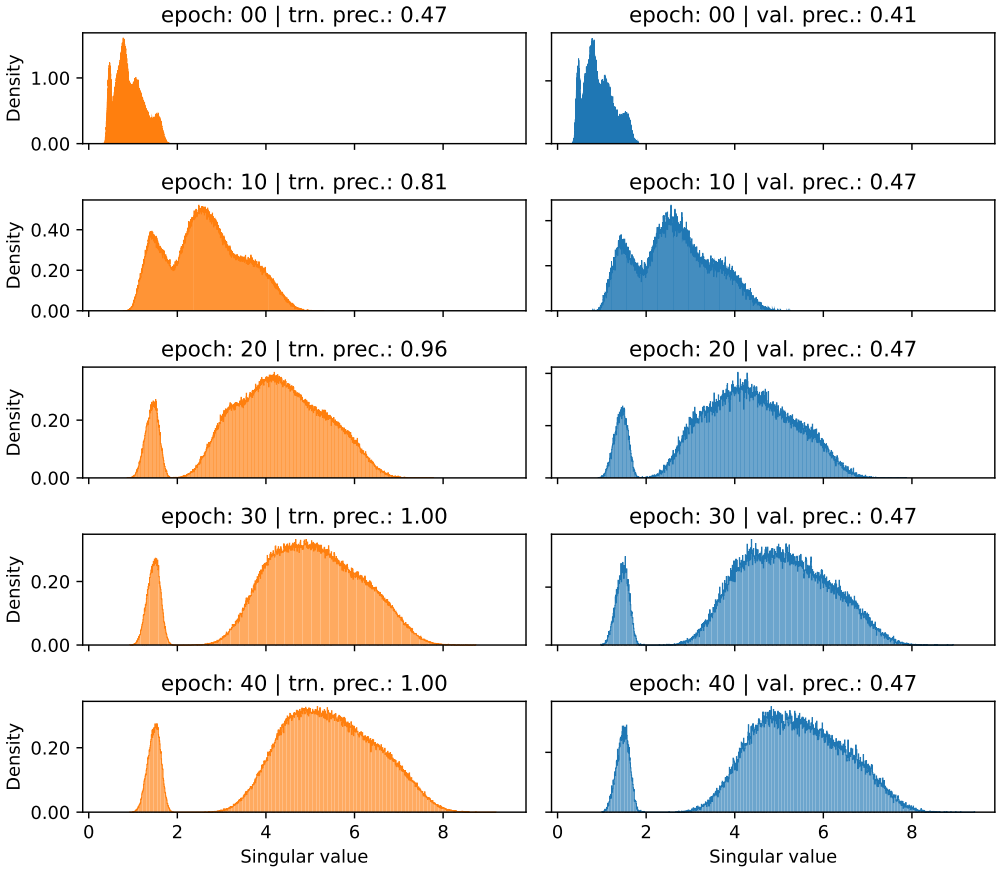


Figure 4.6: Full point train (left) and validation (right) Jacobian spectrum for MLP trained on cifar-10, from first epoch to overfit, with no label noise. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

where we called $v_i^\top \Delta := \Delta_i$. Doing the same for δX and plugging in our objective above, we obtain

$$\begin{aligned}
\max_J \lambda H(J\delta X) - H(J\Delta) &\geq \max_\sigma \left\{ \max_i \lambda H(\sigma_i v_i^\top \delta X) - H(\sigma_1 v_1^\top \Delta, \dots, \sigma_k v_k^\top \Delta) \right\} \\
&= \max_\sigma \left\{ \lambda \max_i \{\log \sigma_i + H(\delta X_i)\} - H(\sigma_1 v_1^\top \Delta, \dots, \sigma_k v_k^\top \Delta) \right\} \\
&\geq \max_\sigma \left\{ \lambda \max_i \{\log \sigma_i + H(\delta X_i)\} - \left(\sum_i H(\Delta_i) + \log \sigma_i \right) \right\}
\end{aligned}$$

Note now that we can do three things to maximize this local lower bound: imagine that we

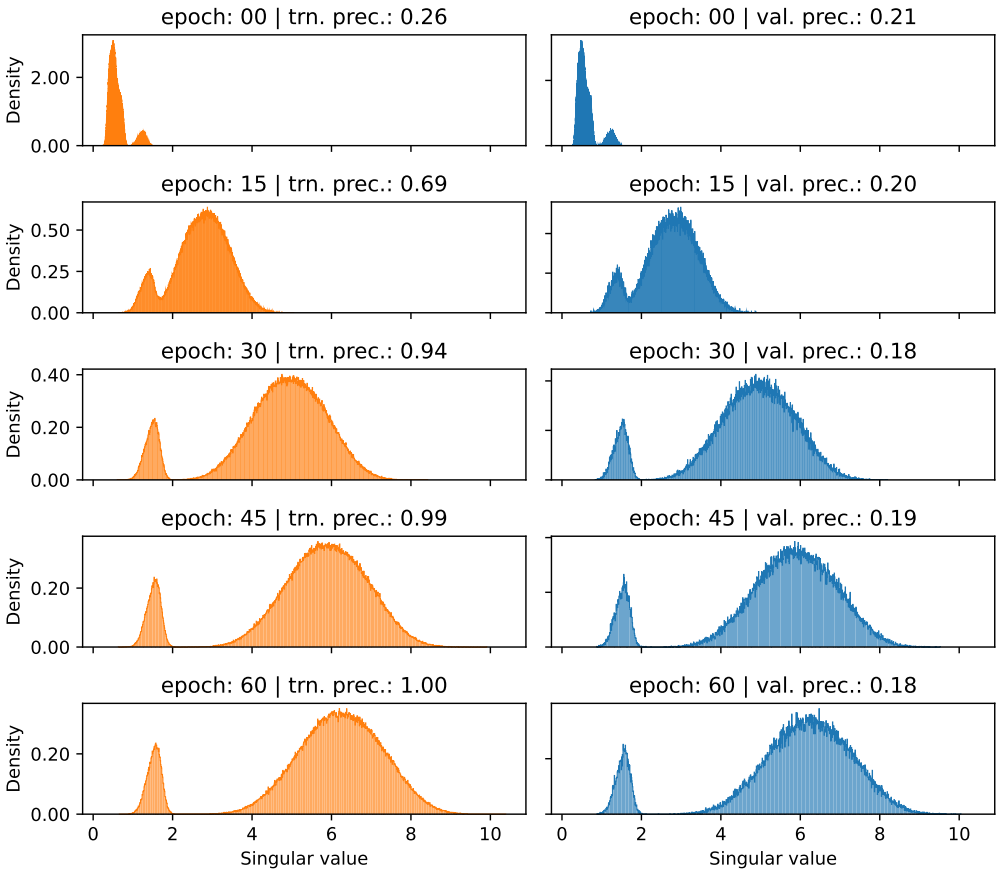


Figure 4.7: Full point train (left) and validation (right) Jacobian spectrum for MLP trained on cifar-10, from first epoch to overfit, with label noise $p = 0.5$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

start training and there is one component of the data that happens to have an image with larger entropy; assuming that the singular values at start are the same, then this is because we are more aligned with the data. And so we will promote this alignment by increasing both the singular value along that direction and rotating it in order to improve the alignment.

The last terms are interesting as well: in order to reduce them and thus increase the lower bound, we can do two things: reduce the mean entropy of the projections of noise onto the singular directions and reduce the mean logarithm of the singular values of the Jacobian. Although the latter can be done without restriction, the former depends on the local shape of noise. Without further assumptions, the only sure way to reduce the mean entropy is to remove dimensions altogether.

But note that since the logarithm can be very negative, it is even better to *keep* the dimensions and just focus on decreasing the singular values to as close to zero as possible.

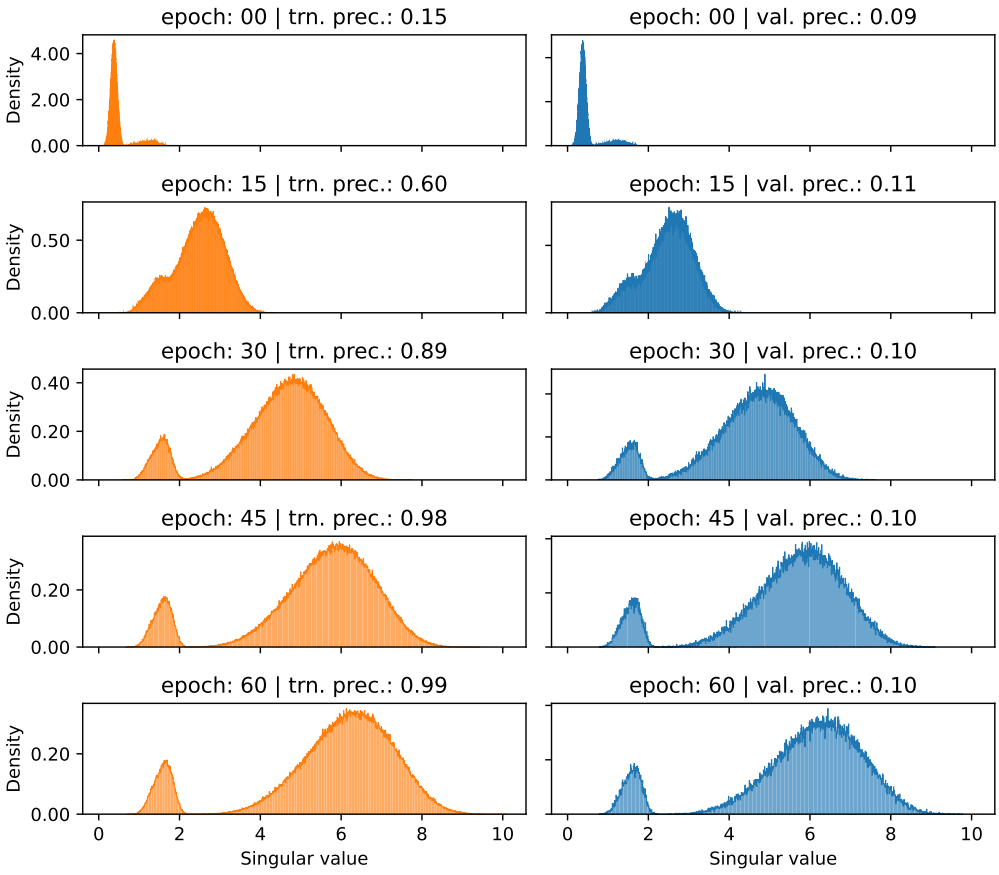


Figure 4.8: Full point train (left) and validation (right) Jacobian spectrum for MLP trained on cifar-10, from first epoch to overfit, with label noise $p = 1.0$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

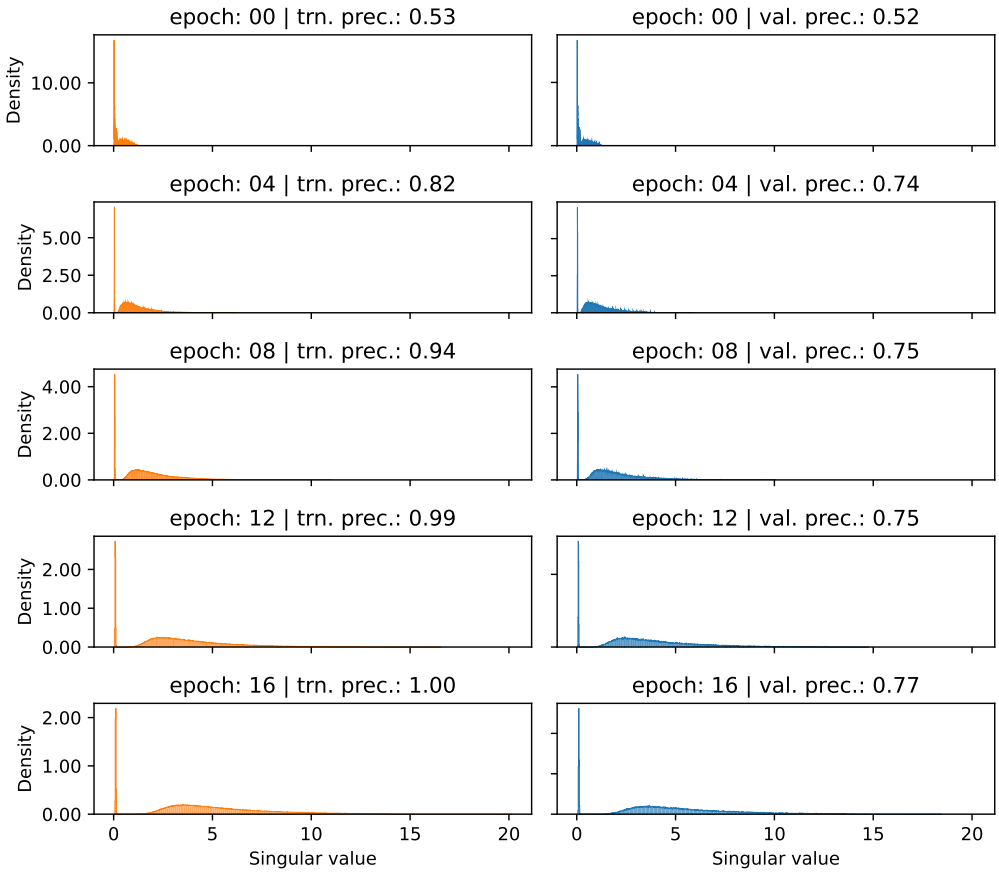


Figure 4.9: Full point train (left) and validation (right) Jacobian spectrum for Alexnet trained on cifar-10, from first epoch to overfit, with label noise $p = 0.0$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

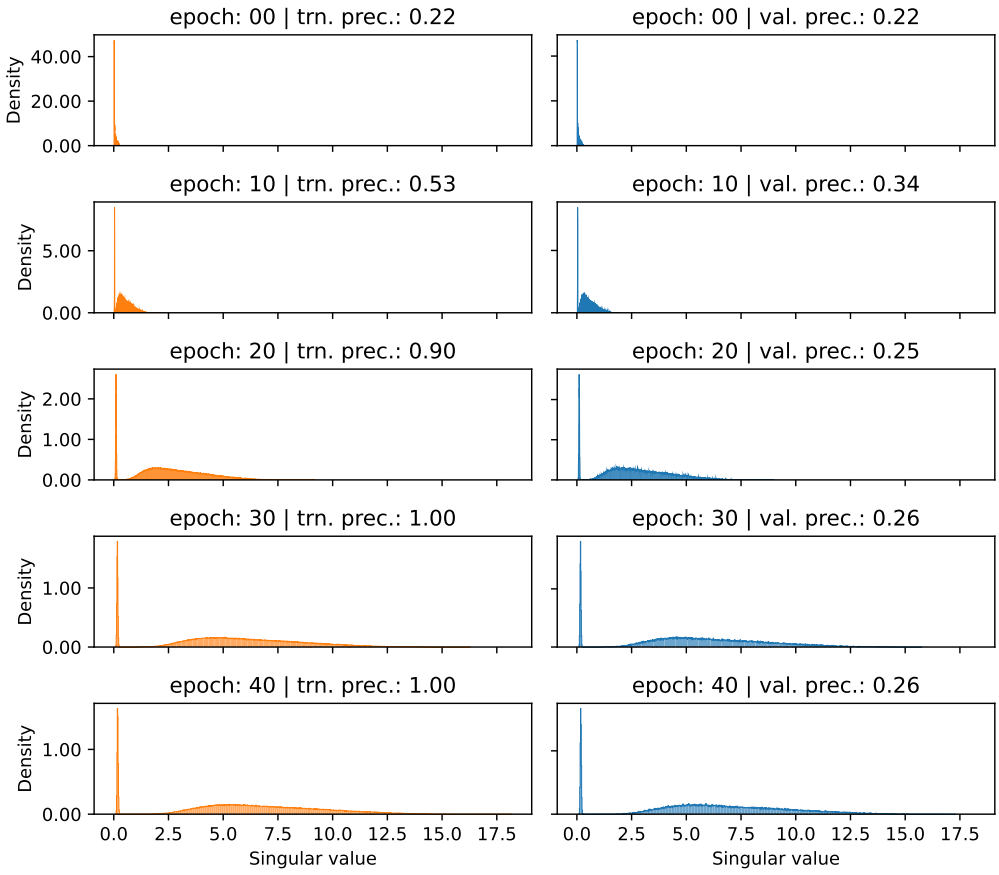


Figure 4.10: Full point train (left) and validation (right) Jacobian spectrum for Alexnet trained on cifar-10, from first epoch to overfit, with label noise $p = 0.5$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

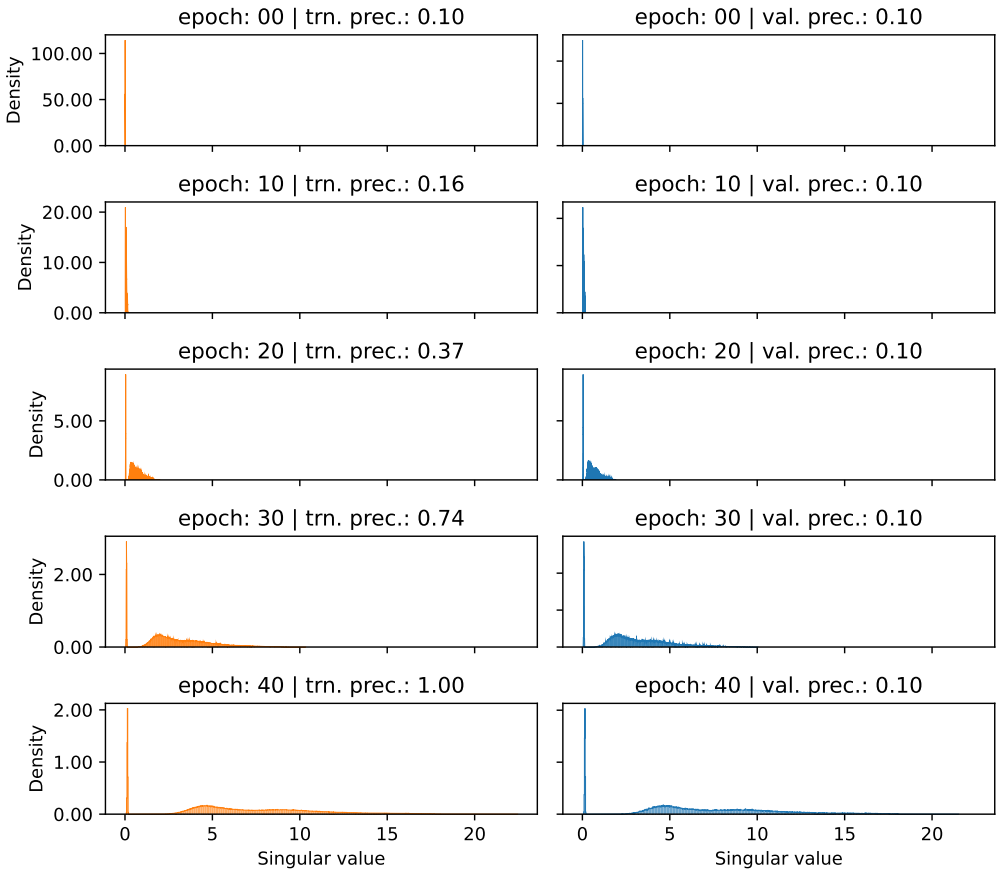


Figure 4.11: Full point train (left) and validation (right) Jacobian spectrum for Alexnet trained on cifar-10, from first epoch to overfit, with label noise $p = 1.0$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

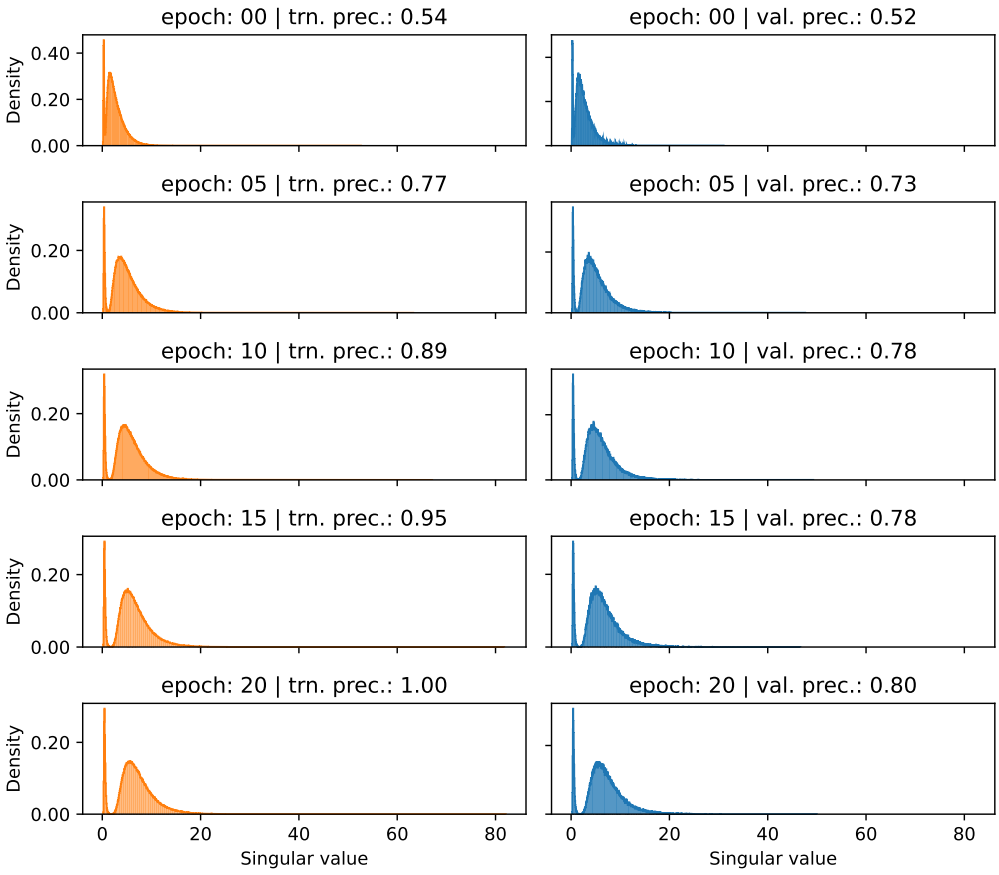


Figure 4.12: Full point train (left) and validation (right) Jacobian spectrum for Inception trained on cifar-10, from first epoch to overfit, with label noise $p = 0.0$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

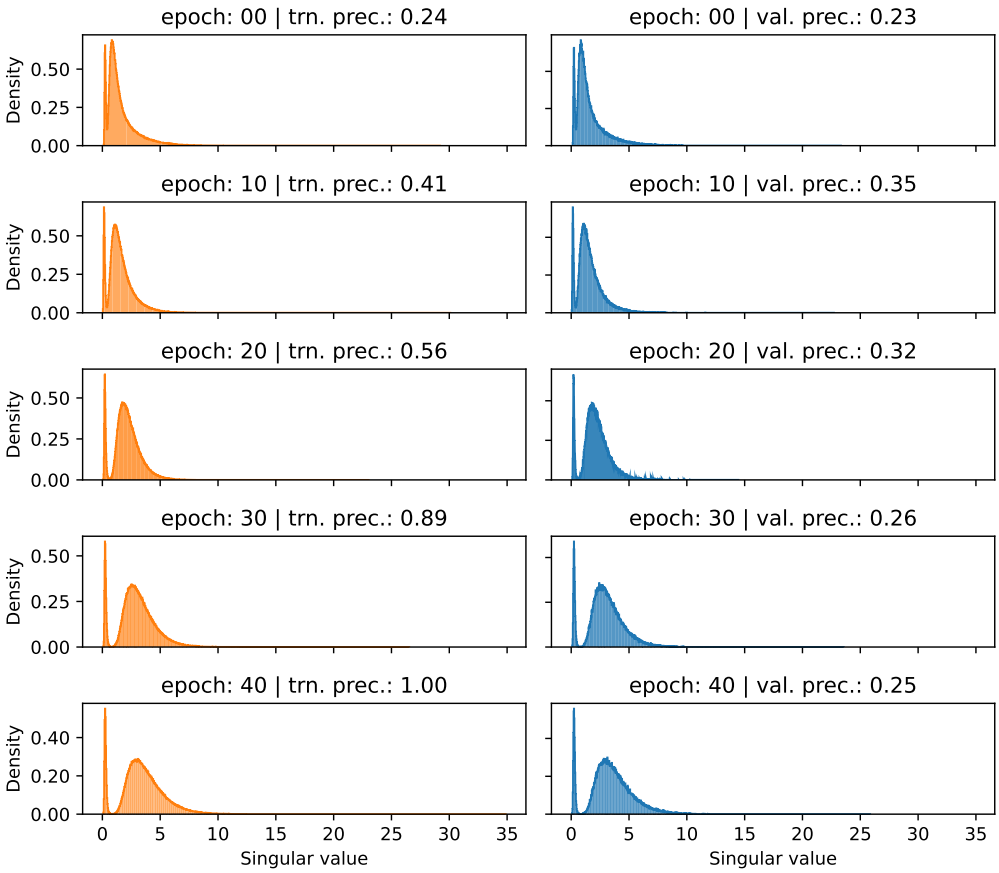


Figure 4.13: Full point train (left) and validation (right) Jacobian spectrum for Inception trained on cifar-10, from first epoch to overfit, with label noise $p = 0.5$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

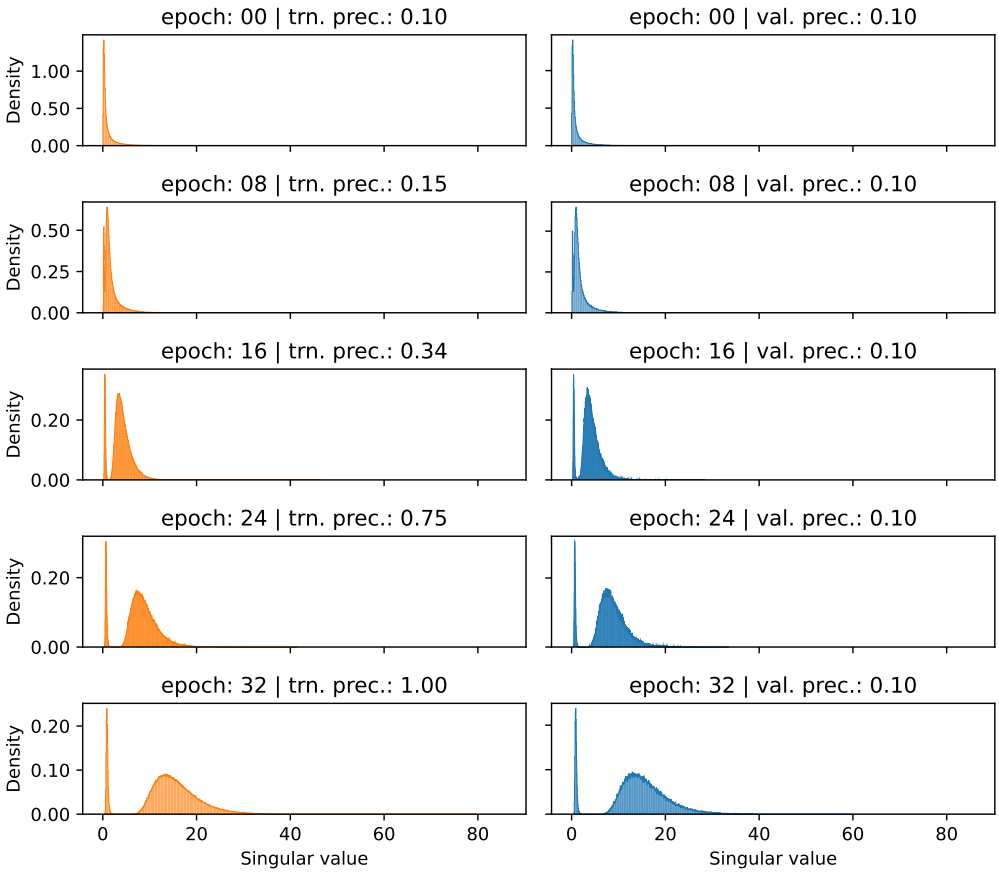


Figure 4.14: Full point train (left) and validation (right) Jacobian spectrum for Inception trained on cifar-10, from first epoch to overfit, with label noise $p = 1.0$. Horizontal scale is the same across all plot in the same column. Epochs and performance are indicated on top of each graph.

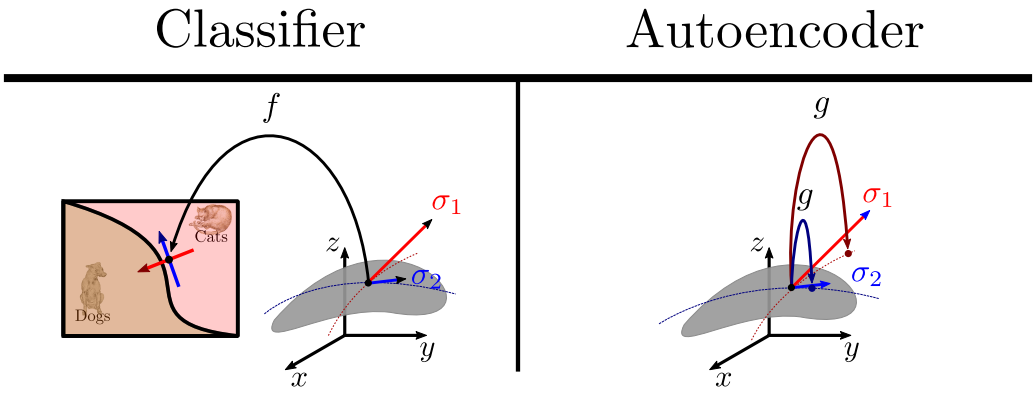


Figure 4.15: Principal directions of the point Jacobian for a classifier f and an autoencoder g for three-pixel pictures of cats and dogs on the neighborhood of a given point. By definition, the norm of the change of the image through f for perturbations in first singular direction σ_1 is maximal among all directions, and similarly for the second direction in the orthogonal space to the first. Note that since the destination space is in $\mathbb{R}^2$, there are only two singular directions in the original $\mathbb{R}^3$. For each point P the two directions are the directions of respectively maximum and minimum change with respect to "cat-dog". As for the autoencoder, reconstruction is much more sensitive to perturbations along σ_1 than σ_2: changes along the latter are reconstructed as being the same image, which means that the model considers them as being noise.

Chapter 5

Conclusion and perspectives

This thesis is about physics guided machine, to which we contributed along two main directions: in the first core of contributions, we worked on integrating partial physical knowledge in the form of a partial differential equation (PDE), specifically the Swift-Hohenberg (SH) equation, to solve the inverse problem of predicting nanoscale patterns in femtosecond irradiated surfaces. In this first direction, we were able to show that in the case of a self-organization process, the dual inverse problem of estimating state and equation parameters simplifies by choosing a feature transformation in the image space in which the initial conditions play a less important role. In the case where data is few and not time-series and the physical knowledge is only partial, this transformation can neither be learned nor derived: we use as transformation the higher-order features of a Convolutional Neural Network pre-trained on a large dataset for a broad task. We proposed a principled approach to choosing such a feature transformation, and an expert based quality measure of the features as well.

We integrated the PDE knowledge by implementing a fast and accurate second-order pseudospectral solver of the SH equation and then by using a great number of pre-generated solutions to learn a surrogate in feature space, on the one hand, and to label the few experimental data with SH parameters of the nearest neighbors in feature space, in the other. This technique allowed us to learn the relationship between laser parameters and SH parameters (with which novel laser patterns can be generated via the solver), a relationship that can be used as an experimental tool to guide new pattern discovery.

In the second core of contributions, we addressed the problem of measuring the amount of knowledge in data and in a (e.g. physical) model. We did so through the angle of the Minimum Description Length (MDL) principle, in the framework of which an elegant solution was found: the so-called one-part MDL, which uses Normalized Maximum Likelihood as detailed in Section 4.3.5. This solution, however, suffers number of technical shortcomings which hinder its application, namely in task-oriented setting. To address these difficulties, we proposed an MDL principle that implicitly defines full model complexity (model + data) in terms of signal and noise as defined by the classification task: *choose the model whose representation of the data can be used to compress the signal, but not the noise*. We show that models driven by this principle locally maximize sensitivity to the signal and minimize the sensitivity to noise, and predict that the point Jacobian spectrum of neural networks trained by gradient descent follow either a power law or a lognormal distribution. We provide experimental evidence supporting this prediction, hinting that neural networks trained by gradient descent are driven by the MDL principle.

Learning Complexity to model Self-Organization of Matter – conclusions and outlook
Our work in this direction led us to make a number of observations, which motivate future research:

(i) First, in spite of the good agreement between the partial SH model predictions and experimental data, we also found evidence that there is more than one SH process at play. This leaves the door open to either using a generalized SH model that integrates several length scales, or to exploring possibilities to combine multimodal single-scale data into a superimposed solution. These two directions are independent, as our original proposed mechanism for pattern formation in Sec. 3.2.4 seems to indicate. On the one hand, multimodality is predicted to exist for large enough fluences, and so should absolutely be incorporated in an improved model. As for using a generalized model, the matter is slightly more complex. If one believes that Rayleigh-Bénard convection is at the origin of pattern formation, then the Swift Hohenberg equation appears naturally. Either the sign-symmetric version in [SH77], in the Boussinesq approximation, or the generalized model used in our work with an additional quadratic term (also known as the Haken model) [Hak77], if one admits a compressible fluid (which is indeed a more natural hypothesis), and which is more compatible with the observed hexagonal patterns. As for further options for modeling the hydrodynamics, there is the Kuramoto–Sivashinsky equation [Kur78; Siv77], for example, a notable 4th-order equation which can be derived in a hydrodynamic setting and which has a great variety of pattern solutions [KKP15]. A more involved and alternative approach would consist in modelling the dynamics at different times using different equations. This would allow integrating knowledge and data at several stages of the dynamics, in an iterative approach. This could alleviate the severe data constraints by focusing on building physical models for different parts of the dynamics using available data (absorption, for example); these could later be integrated as "physical knowledge building blocks" that could be used to model the full, complex dynamics.

(ii) Second, we observed that pattern features, as described by the parameters of the SH equation, are not independent; finding new pattern regions requires searching laser parameter space "creatively" by looking at regions of laser-parameter space where some SH parameter shows great variation. This is to be expected, as the SH equation has a limited number of pattern solutions, but poses an important limitation for industrial applications, in which we would like to have maximum flexibility in what kind of patterns can be produced. To address this important issue, note that it is not difficult to leave this constraints even while keeping the SH model, which is known to have solutions that are heavily dependent on the boundary conditions (which we kept periodic throughout this work), for example, and by taking into account pattern interference (although the heavy attenuation of spatial frequencies away from the unstable modes makes interference negligible unless the second hydrodynamic process selects spatial frequencies that are a multiple of the first). The sub-5 nm polishing procedure, perpendicularly polarized double pulse, as well as the 100 alignment of Ni, strongly contribute for the symmetry setting in which Rayleigh-Bénard convection can occur. Controllably stepping outside these constraints could open the door to other self-organization convection mechanisms, which could see other pattern solutions. This idea is reinforced by the fact that there are recently observed patterns that do not appear to be similar to known solutions of the SH equation. To do so in a principled way requires iteratively testing different mechanisms for pattern formation, which would then be compared to experimental data, and ultimately be used to propose new experiments — in short, the scientific method, combining principled physical modelling and physics-guided-ML-assisted experimentation.

(iii) Third, we noted that the model learns interesting features from few data even while not extrapolating particularly well. This opens the door to a dialectic approach to novel pattern discovery, even while keeping within the constraints of the SH model: one could acquire new experimental

data iteratively in order to fill-in the gaps in the laser parameter space until the predictions stabilize; since the SH model is already trained and data is pre-generated, integrating new experimental data only requires retraining a simple model on few data. This idea could of course be replicated for other models other than the SH model, to guide the physical exploration of experimental parameter space. The main challenge in our approach is the scarcity of data. Although it is always possible, in principle, to acquire more experimental data, in practice the amount of data is unlikely to change because the cost is too high. In the laser pattern case, one possibility to circumvent this limitation is to combine data from experiments on several materials using domain adaptation [Red+19], which would increase data by an order of magnitude and open the door to exploring patterns in unseen materials, which has great interest for applications.

(iv) For another possible direction of future work, recall that as explained in the schematics Fig. 3.2, the role of the feature transformation is crucial, as it allows us to dramatically reduce the complexity of the problem by reducing the dimension. But it is also crucial, as mentioned in the introduction (1), that the feature transformation incorporates physical knowledge. The choice of a Convolutional Neural Network classifier integrates translation equivariance explicitly, and also other symmetries implicitly which are learned from data. There is clearly work to be done to incorporate other symmetries *explicitely* by learning e.g. by leveraging scattering networks [BM12]) or via [DNT22] or even [CW16]. A learneable differentiable feature transformation that incorporates symmetries can be used in an integrated approach where symmetries and dynamics in projected space could be learned separately, with the latter being learned using, for example [Rud+17; LLD18] or [Xu21], in a physics-guided approach to model order reduction.

Is My Neural Network Guided by MDL? – conclusions and outlook (i) The first direction that we would like to explore regarding our work in this direction is one of systematization: for example, we would like to understand the role of the radii and number of the constant Jacobian neighborhoods (or the error budget, cf. 4.1), which remained largely unexplored in this first work. Another interesting research direction is to relax the manifold hypothesis used in Section 4.4.3 to derive a spectral version of our MDL objective. The predicted effect on the spectrum could be experimentally compared against known geometries. Although we designed our experiments around the notion of "natural" noise, a systematic exploration in an artificial setting could also be useful. Another rather obvious research question is to explore using our MDL objective in learning by integrating in the loss function.

(ii) Another line of research is to understand the connection between our proposal and other measures of complexity and generalization. Specifically, we plan, aiming at a generalization bound, to extend the connection between our MDL formulation and sensitivity established in Section 4.4.2, by making our MDL objective layer-wise as in [Aro+18]. A layer-wise MDL objective would potentially open doors to comparisons with the Information Bottleneck Principle [Sax+19] which would be interesting to explore. Another possible direction for drawing from our MDL objective to obtain generalization bounds is to use the PAC-Bayesian Theory as proposed in [Via+23] to derive bounds from arbitrary complexity measures (inspired from empirical results presented in [Jia+19]). These bounds could then be compared with those obtained via other methods. Another possible extension is to use our findings to explain the power law behavior of the spectra of the layer weight matrices and connection to generalization gap found in [MM18; MM20] in the context of Random Matrix Theory, where the authors show that the spectrum of the layer weight matrices of Neural networks with published weights (thus that have sufficiently good performance to merit publishing) follow a power law behavior, and propose a measure of generalization for pretrained neural networks.

To make the connection with our work we note that each point Jacobian of ReLU networks is a sub-matrix of the product of the network weight matrices, which can be expressed in terms of the singular values of the point Jacobian submatrix via an interlacing inequality [Tho72].

(iii) To make our MDL formulation more general, we would like to extend it to a regression setting, as this is the context of interest in a number of problems of physical interest. This should be possible in principle, since the MDL principle that we proposed is applicable to settings where a meaningful definition of signal and noise in terms of the task exists. In a regression setting, this is also the case. But in a regression problem, the number of output dimensions is typically much higher, which poses considerable practical difficulties in computing the singular values of the Jacobian.

(iv) Finally, as stated in the introduction (1), MDL provides a natural formalism in which to compare on the same footing knowledge in data, and knowledge in some physical information – which comes in so many different shapes and forms. As an architecture-independent, formalism independent way to assess the task-relevant knowledge in data and in physical information, MDL is arguably the ideal setting to compare models incorporating physical information and data. Another possible direction for future research is thus to apply our MDL measure to problems where physical information is available, in order to find, as we mentioned in the introduction (1), the "data-equivalent of physical knowledge". To do so, we would follow a systematic approach that explores a range of physical problems, where physical knowledge is formulated in different ways (e.g. a PDE governing the quantities of interest or a derived quantity, symmetries, conservation laws), and examine the behavior of trained models with respect to the introduction of data, and of knowledge.

Taylor entropy We introduced Taylor in Section 2.6.6 as a measure of complexity of dynamical systems that is applicable in the multidimensional setting. This measure, as can be seen in e.g. Figures 2.34 and 2.36 provides a good measure of complexity that is applicable in a very noisy setting. As such, this measure merits further investigation on its own: for once, one would like to study it in a more controllable setting (with PDE-generated data, for various noise settings) and in different dimensions. In an true time series setting, instead of the averaging procedure that we used for the SEM field, where we only have access to two-dimensional sign patches, one would have access to three-dimensional sign right rectangular prisms. A measure of diversity of these prisms bares a number of wq resemblances to permutation entropy, with derivative sign sequences in the former taking the role of ordinal patterns in the latter. One could also examine the complexity of each derivative individually, and its relationship to the form of the (known) governing equation.

Appendix A

Appendix

A.1 Experimental section: full figure list

In this section, we show the application of the aforementioned measures of complexity to (i) a panel range of femtosecond laser induced patterns with constant N and varying laser fluence and time delay between pulses and (ii) five different increasing N series (cf. Figure A.1).

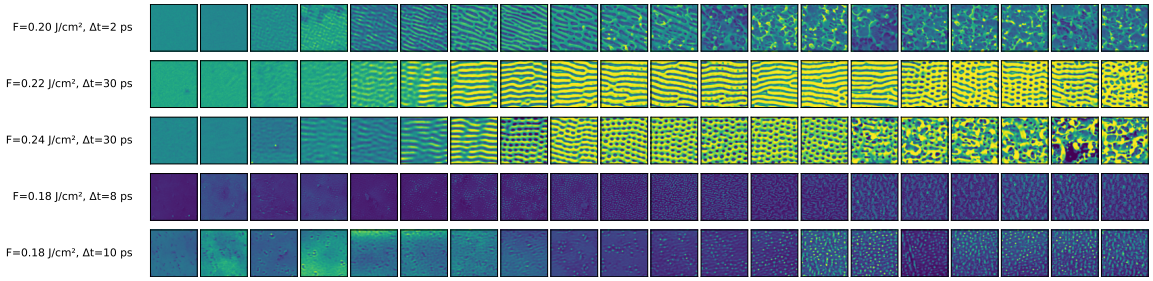


Figure A.1: Visualization of SEM (Scanning Electron Microscopy) field samples for different laser parameters. Each row represents a series of samples generated with specific laser parameters, as indicated by the y-axis labels, at increasing values of N. Each column represents a sample taken at equal intervals within the series.

A.1.1 Gray levels complexities

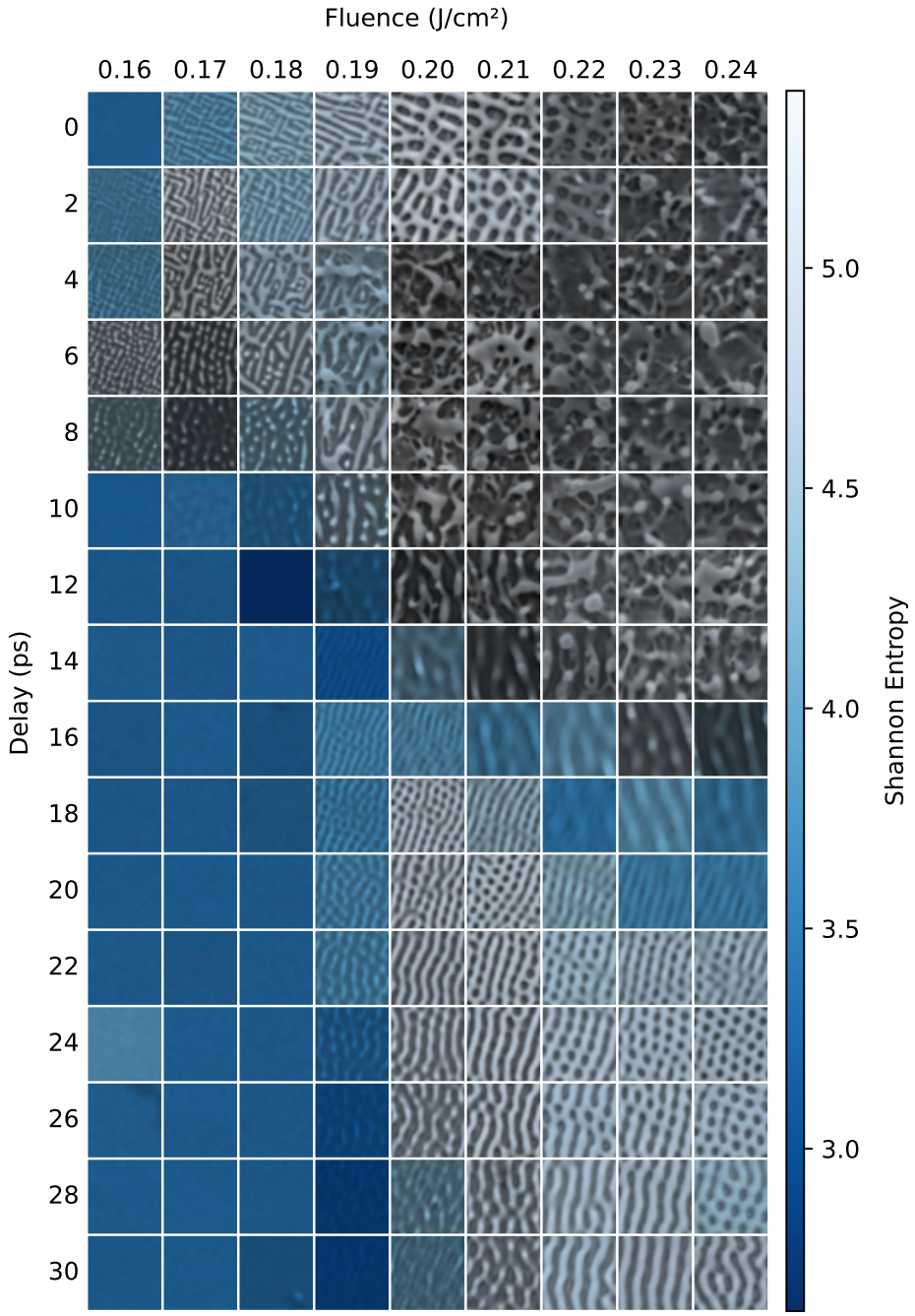


Figure A.2: Shannon entropy of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

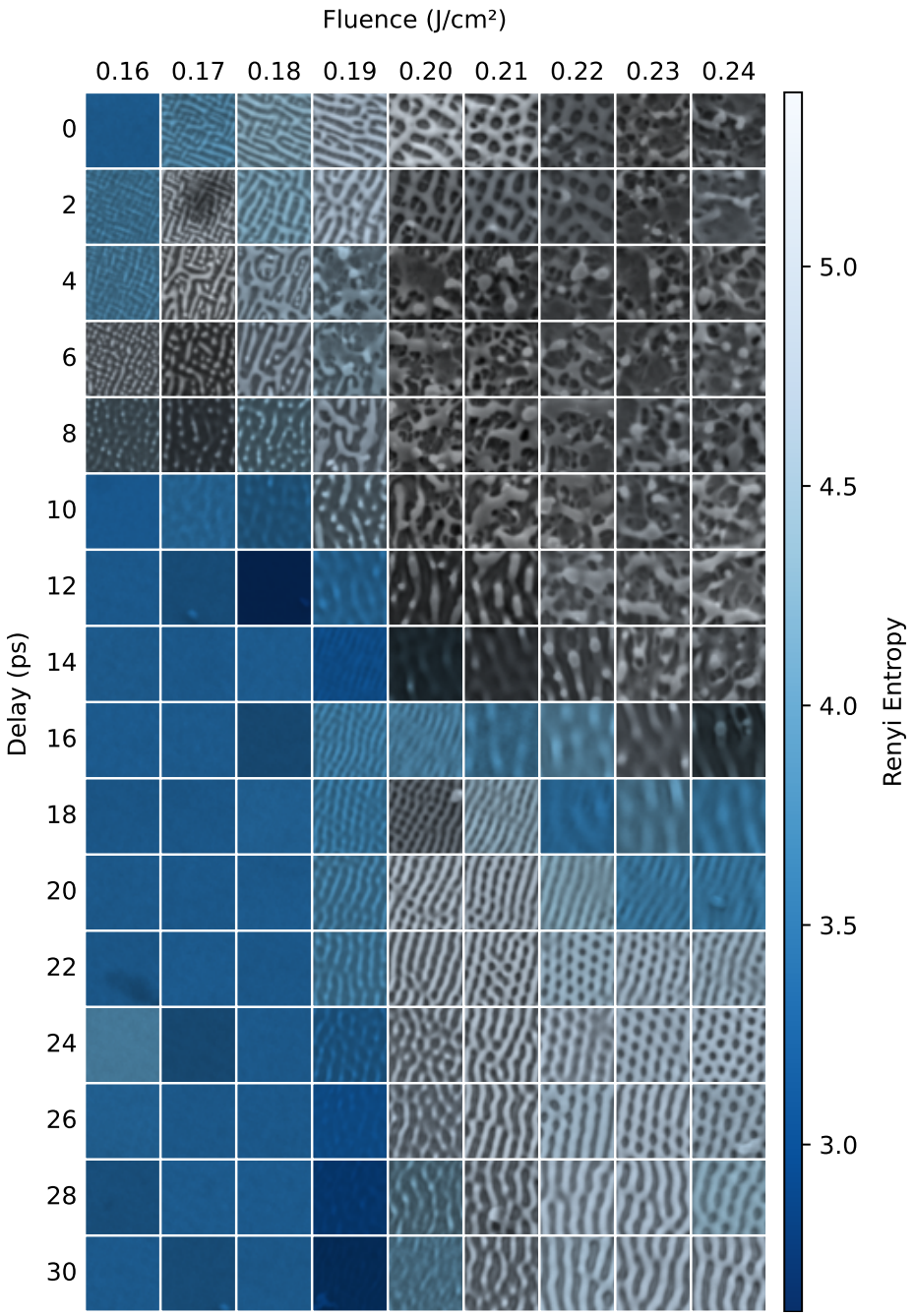


Figure A.3: Rényi entropy of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

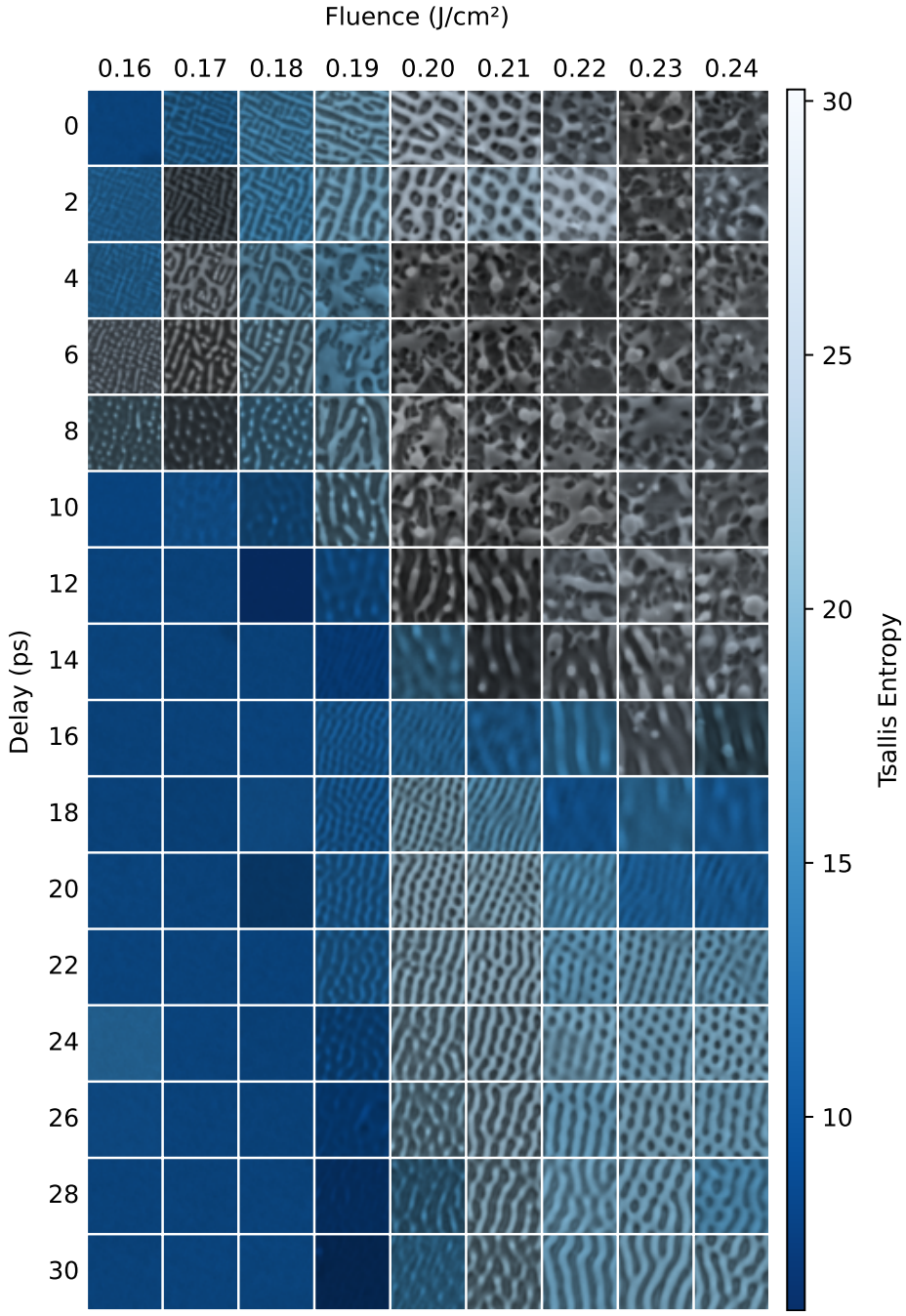


Figure A.4: Tsallis entropy of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

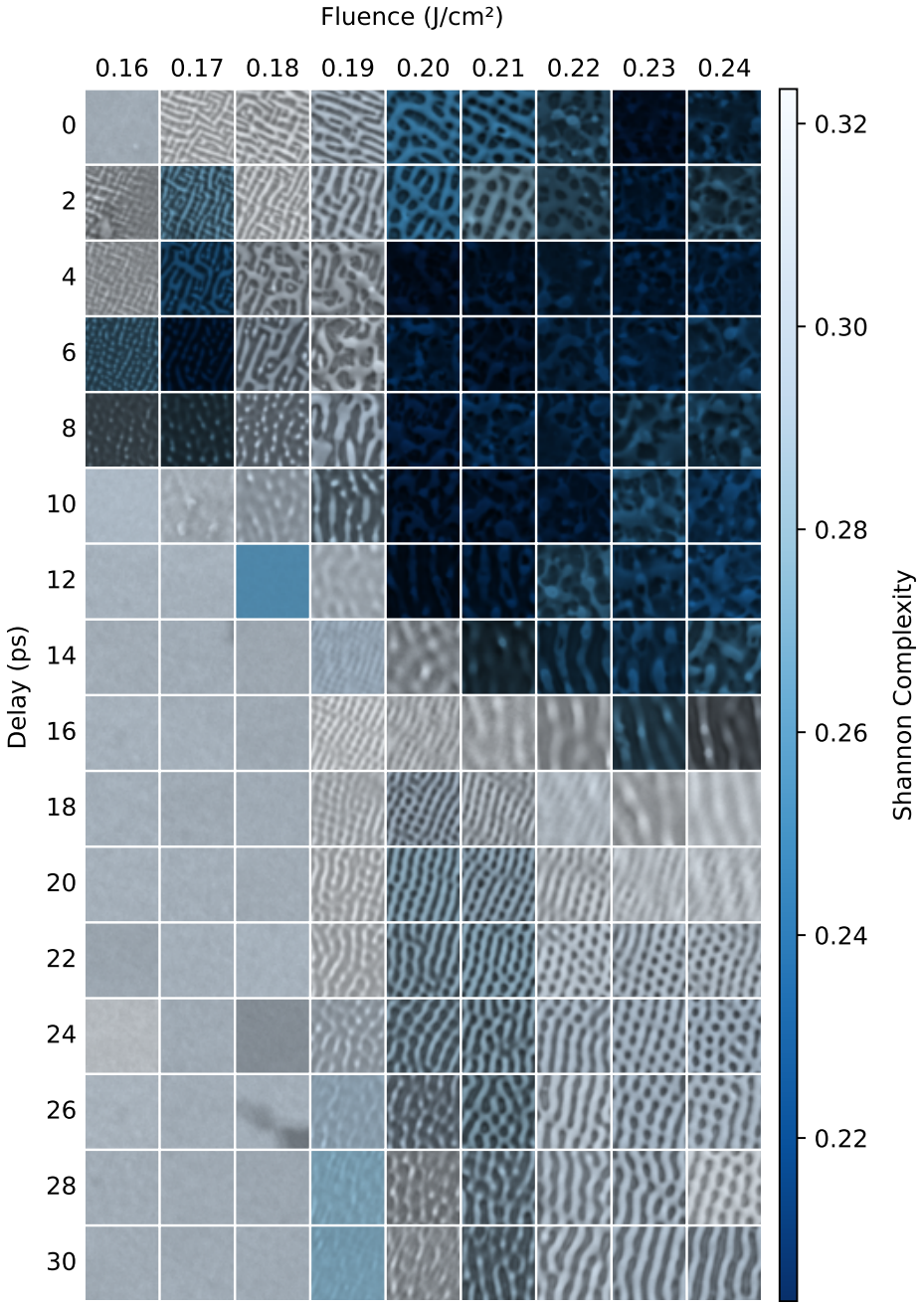


Figure A.5: Shannon complexity of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

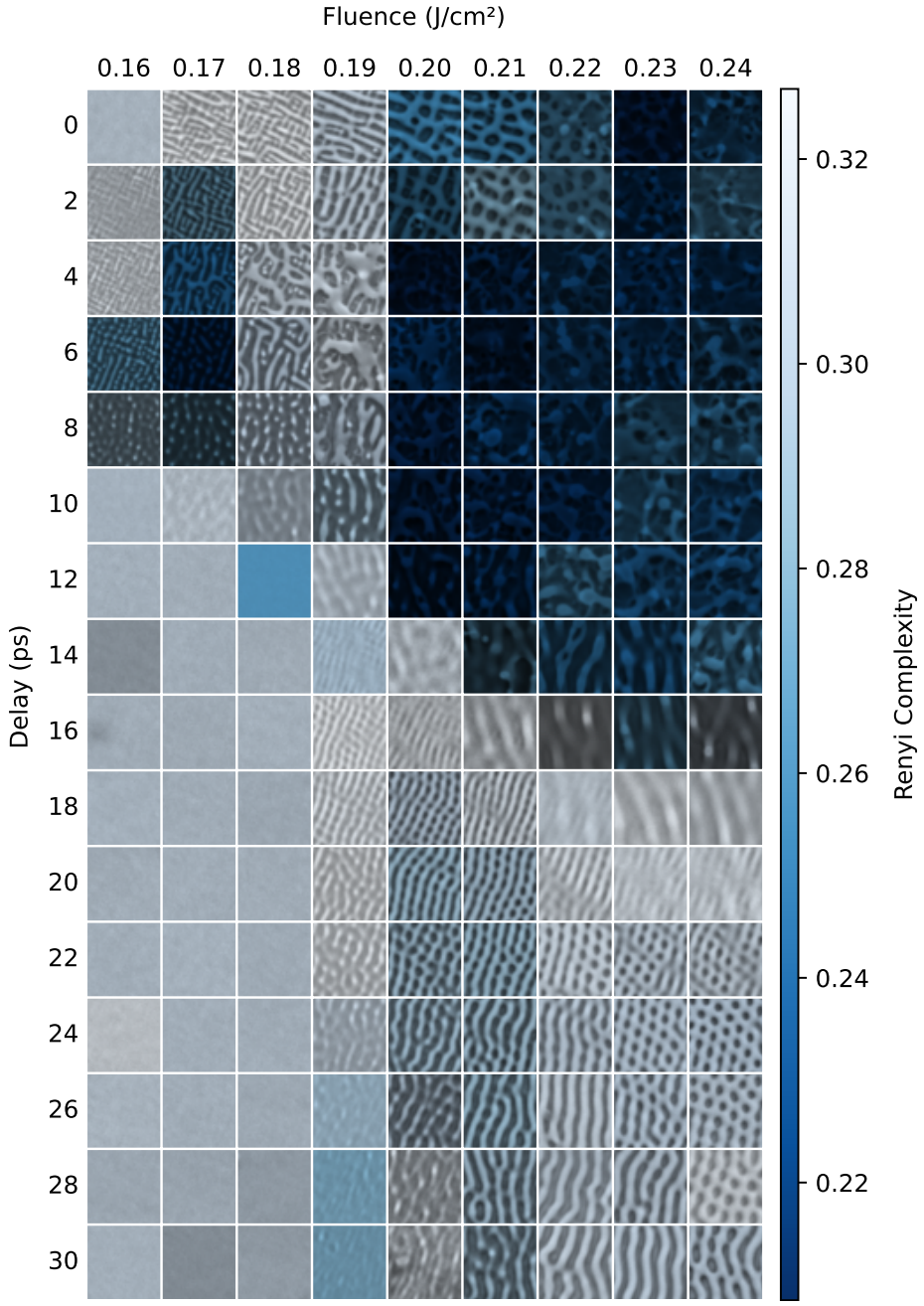


Figure A.6: Rényi Complexity of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

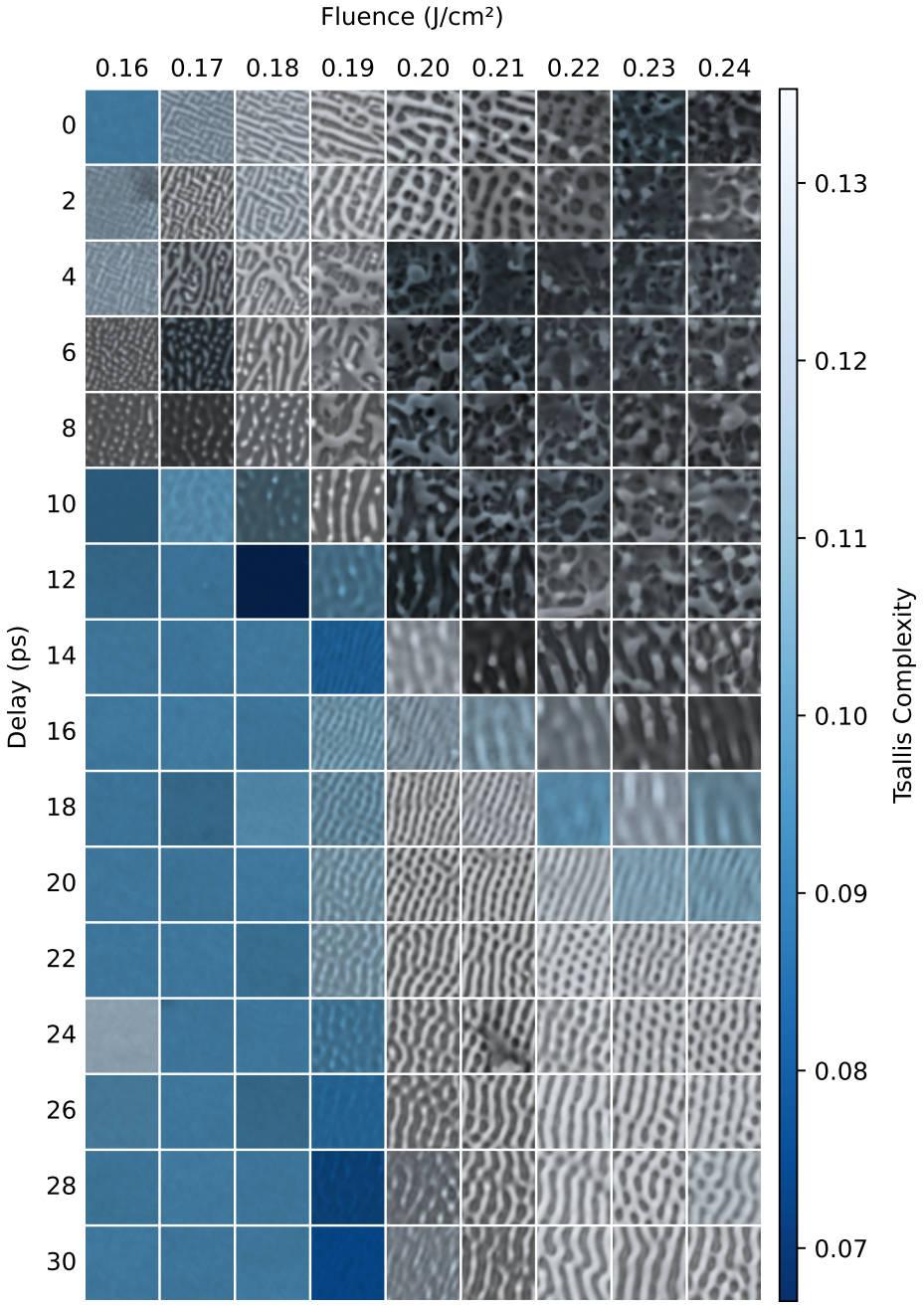


Figure A.7: Tsallis Complexity of the gray levels of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

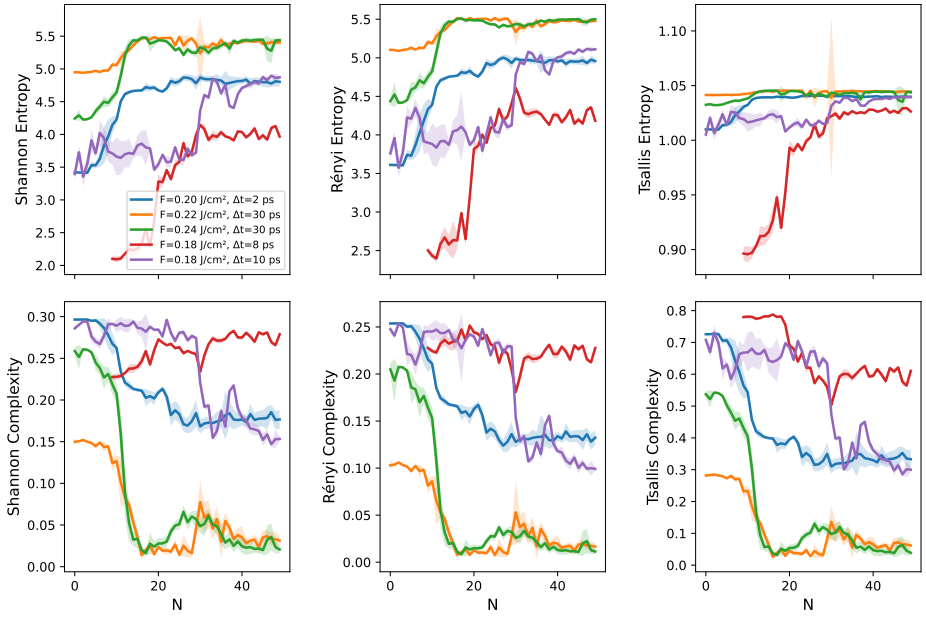


Figure A.8: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) gray level entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image. All entropies increase up to a certain value of N, where they stabilize. The exception being the Purple series ($F = 0.18$ J/cm², $\Delta t = 10$ ps), which presents a great variety of different structures that form during the dynamics, ranging from holes to hexagons to humps and chaos

A.1.2 Gray level runs complexities

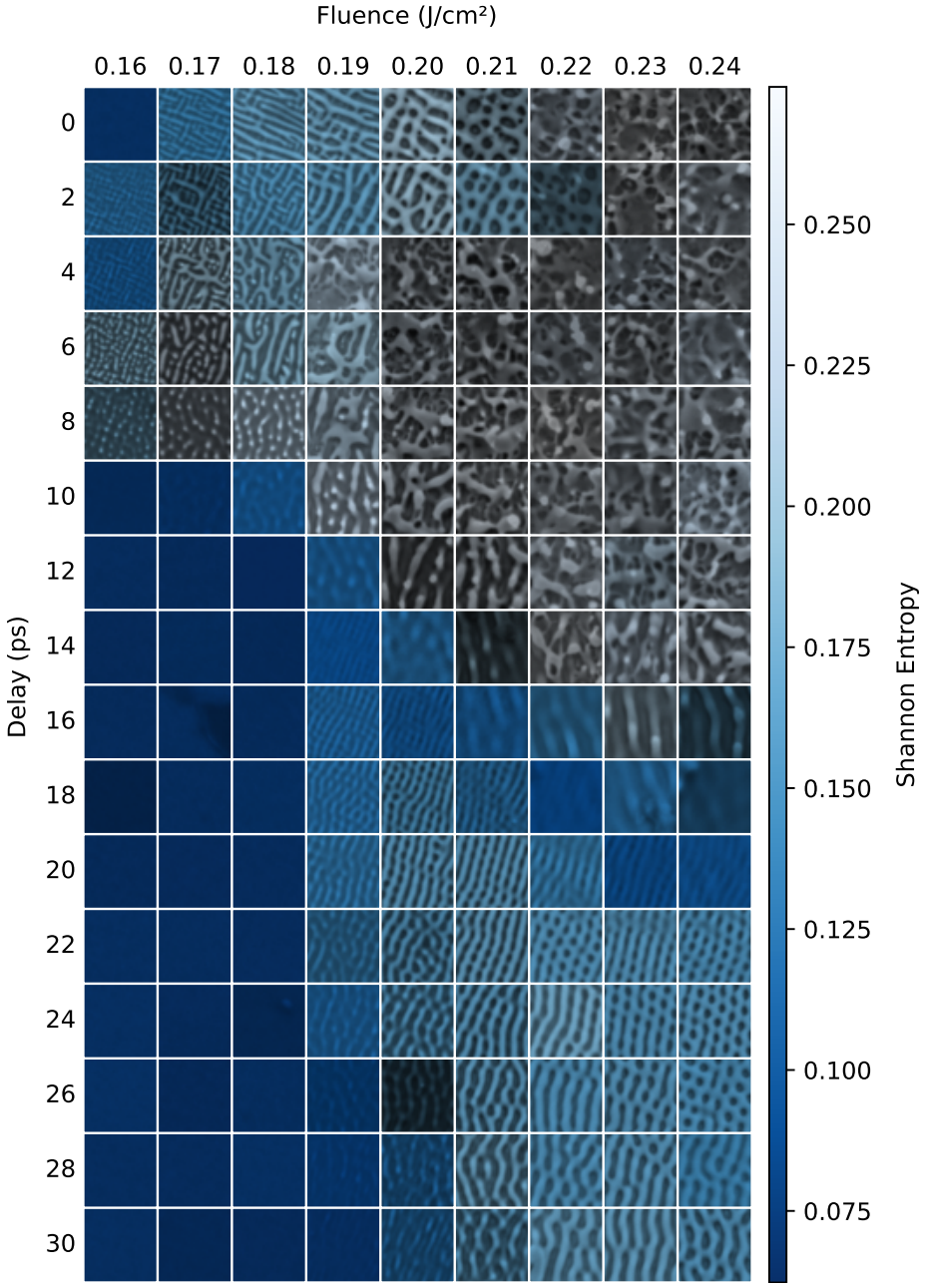


Figure A.9: Shannon entropy of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

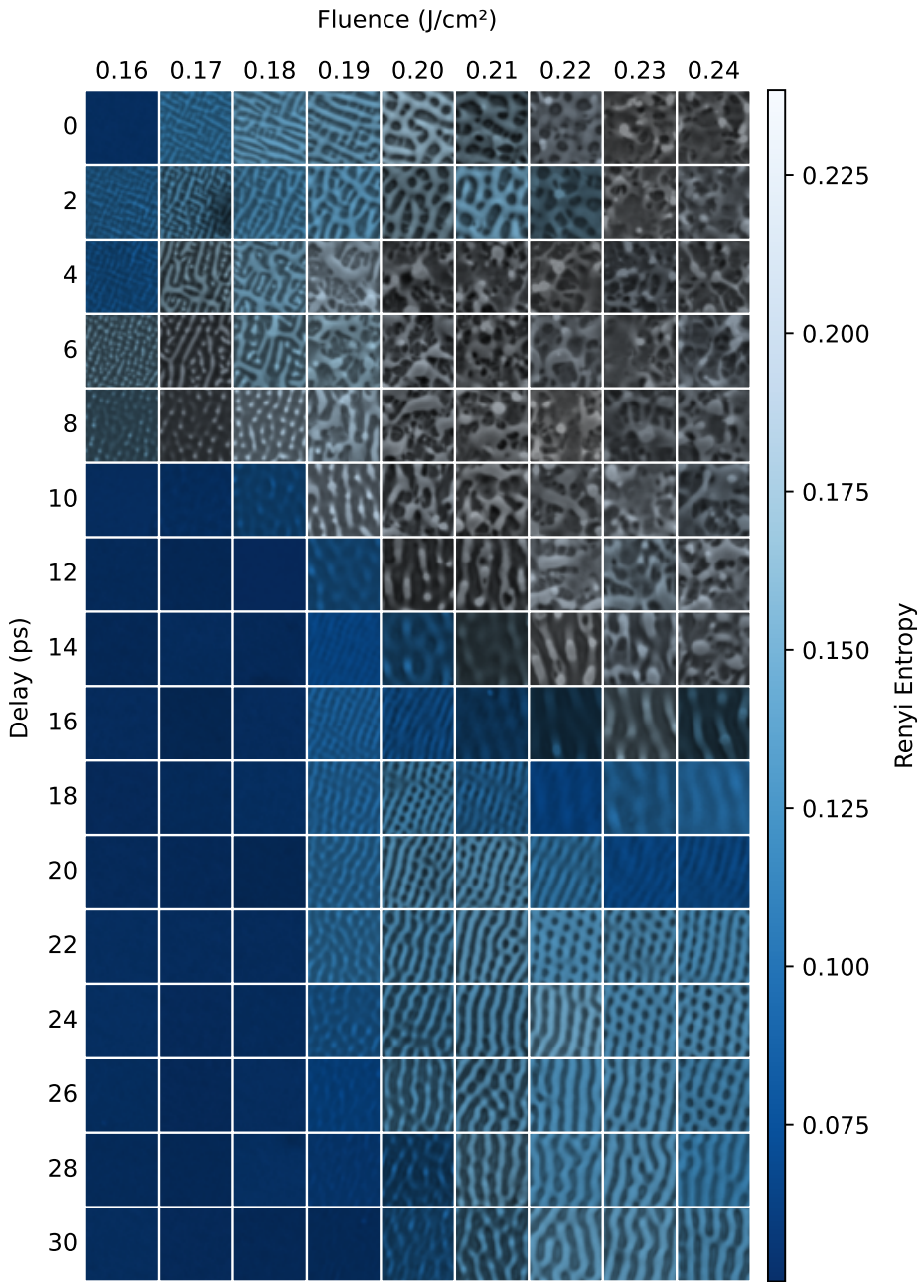


Figure A.10: Rényi entropy of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

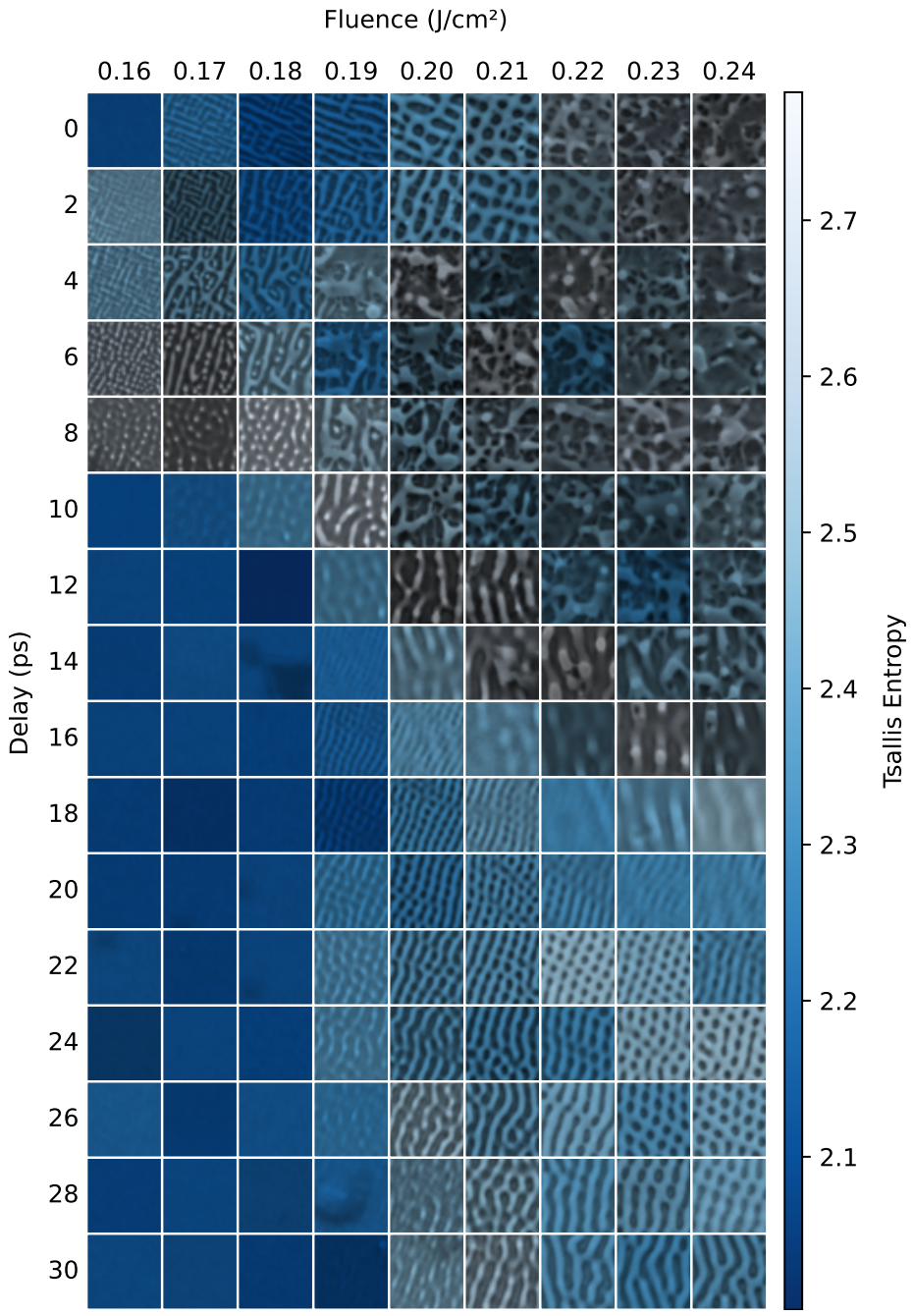


Figure A.11: Tsallis entropy of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

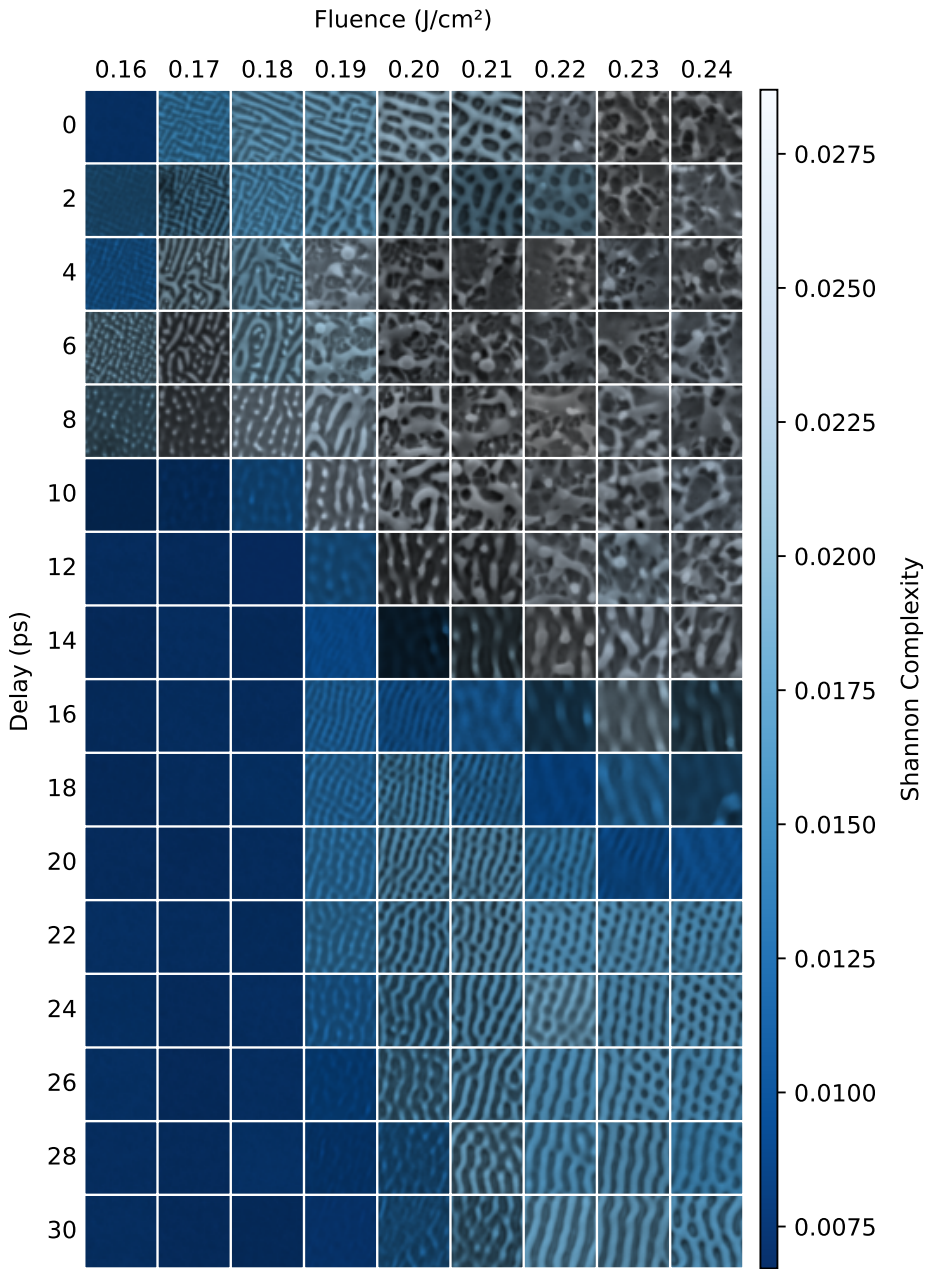


Figure A.12: Shannon complexity of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

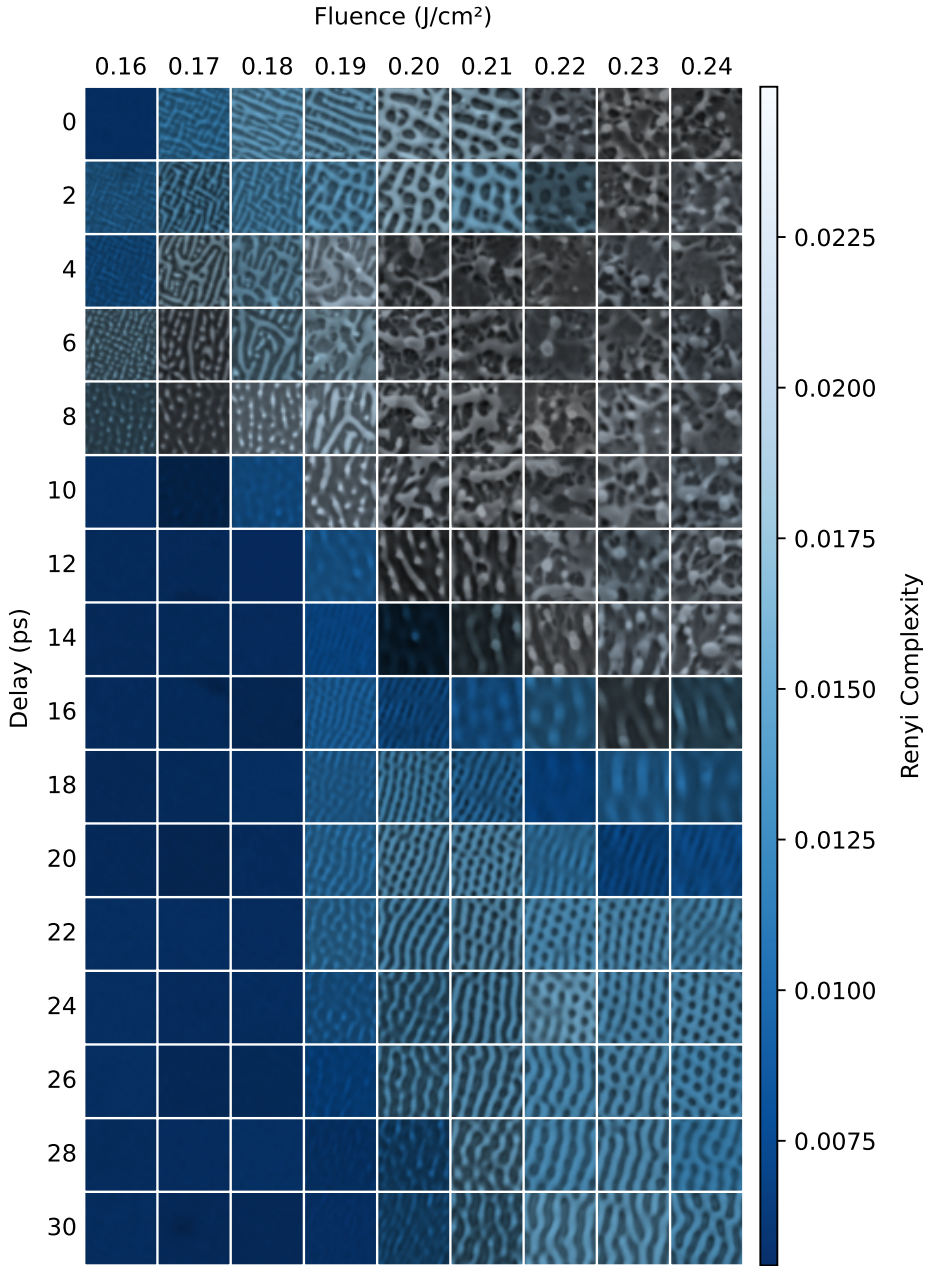


Figure A.13: Rényi Complexity of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

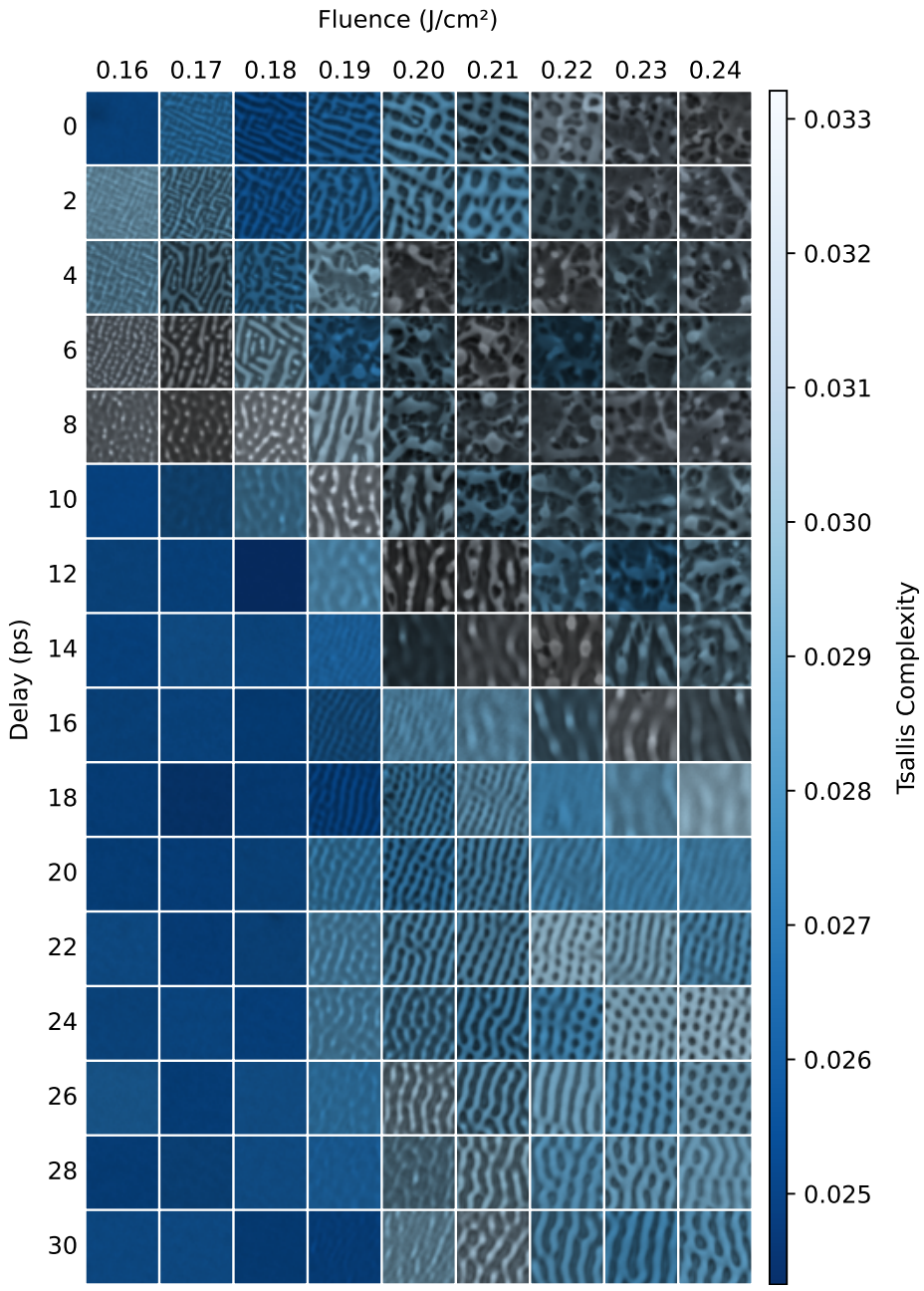


Figure A.14: Tsallis Complexity of the gray level runs (glr1) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

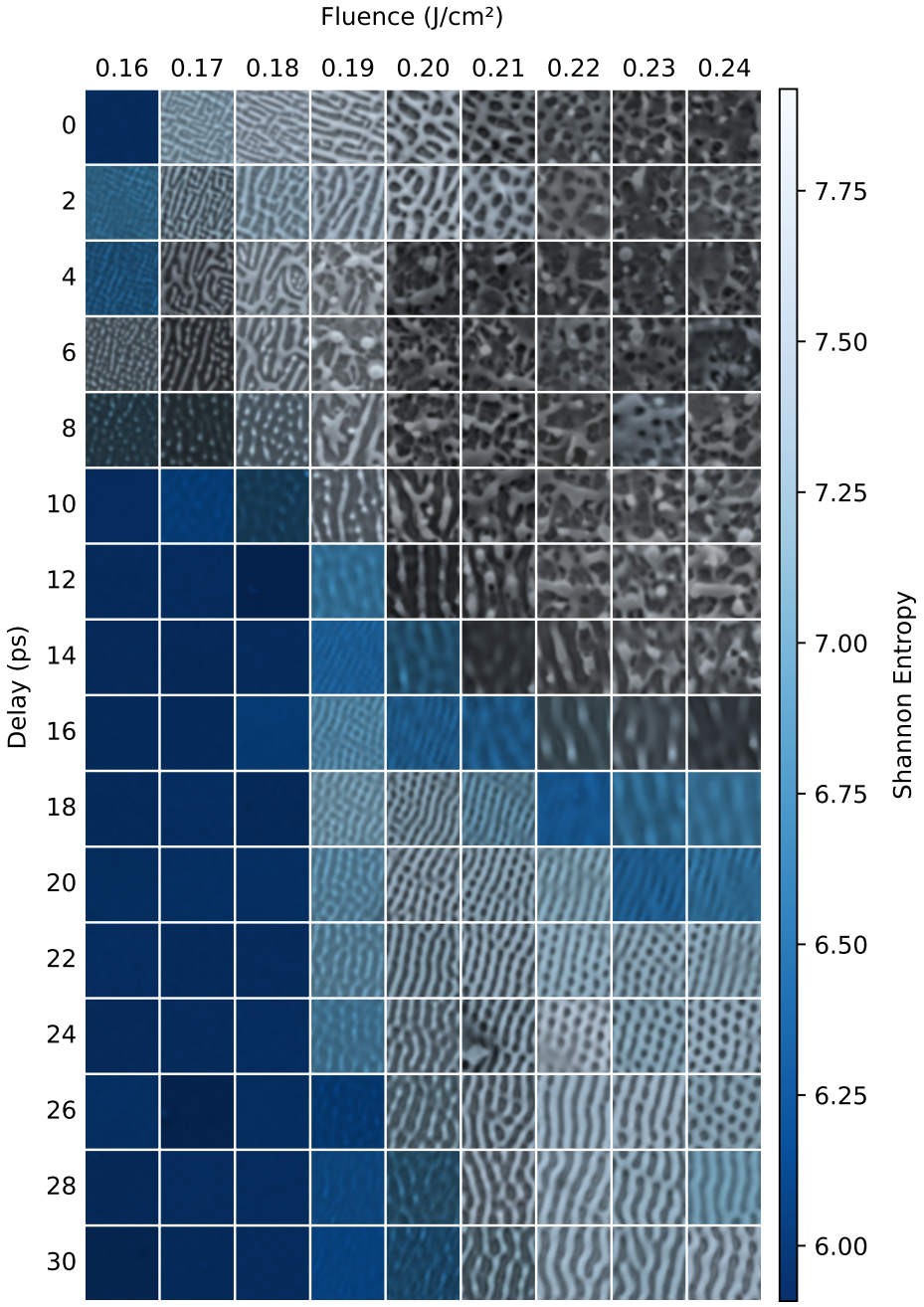


Figure A.15: Shannon entropy of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

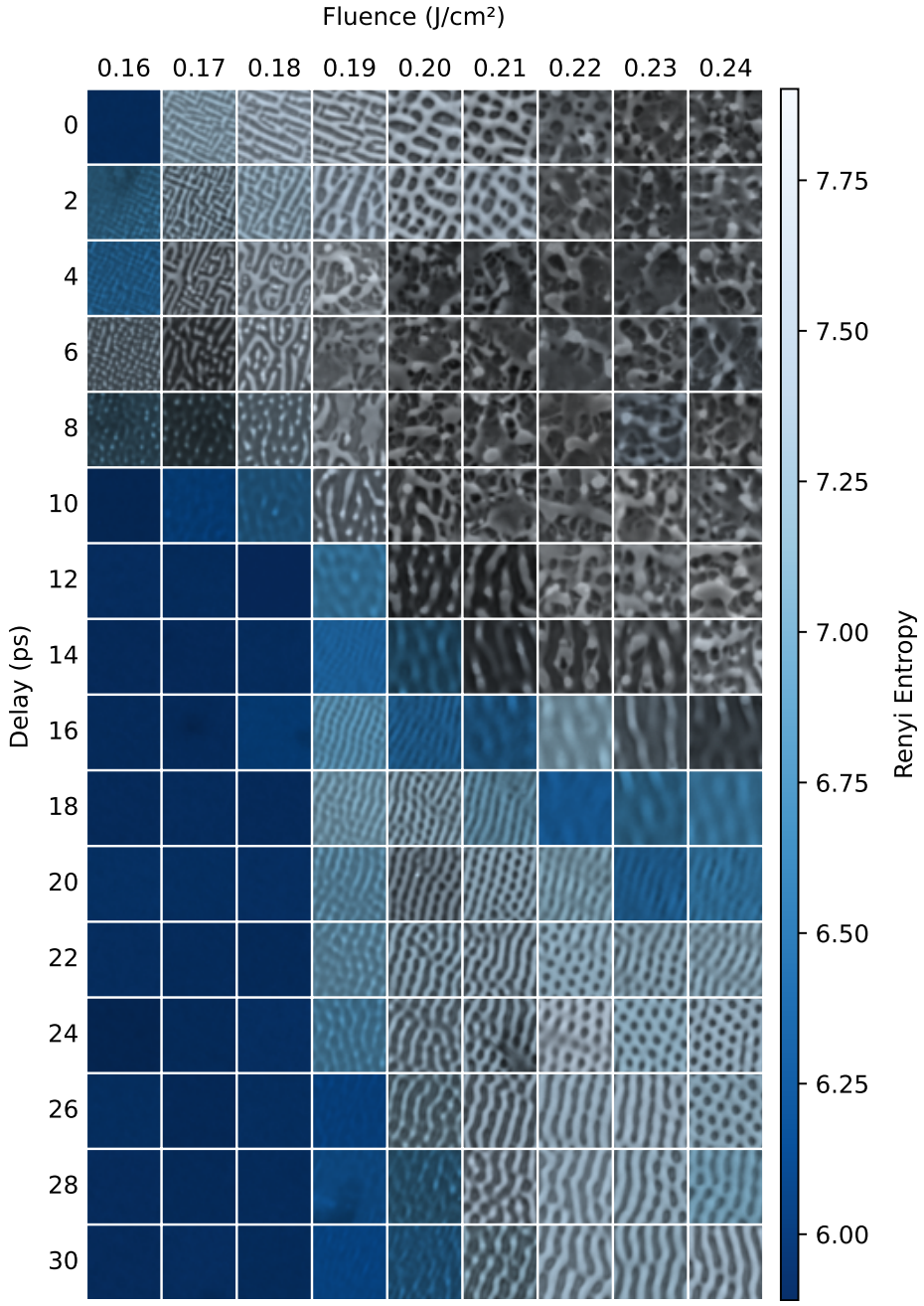


Figure A.16: Rényi entropy of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

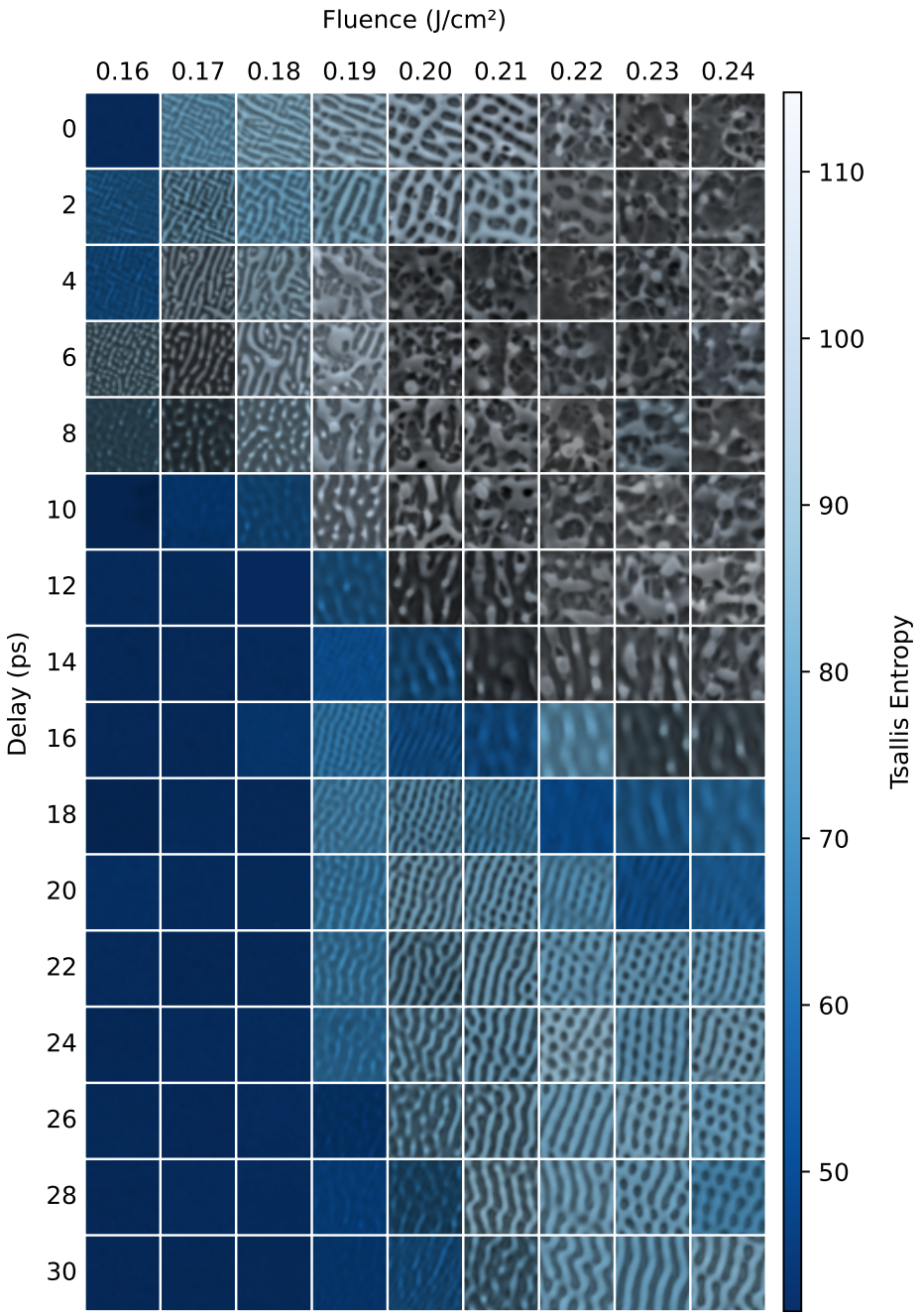


Figure A.17: Tsallis entropy of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

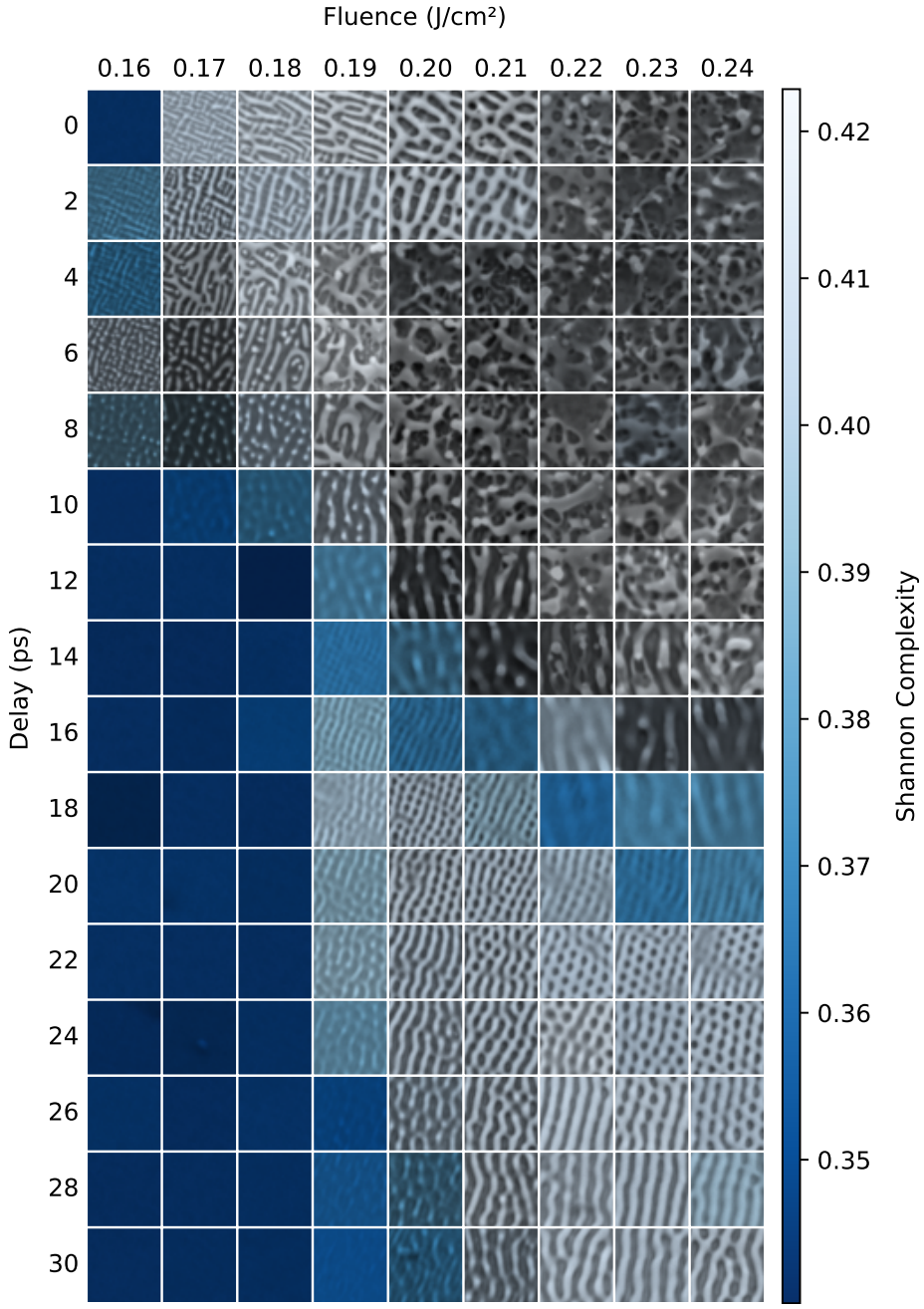


Figure A.18: Shannon complexity of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

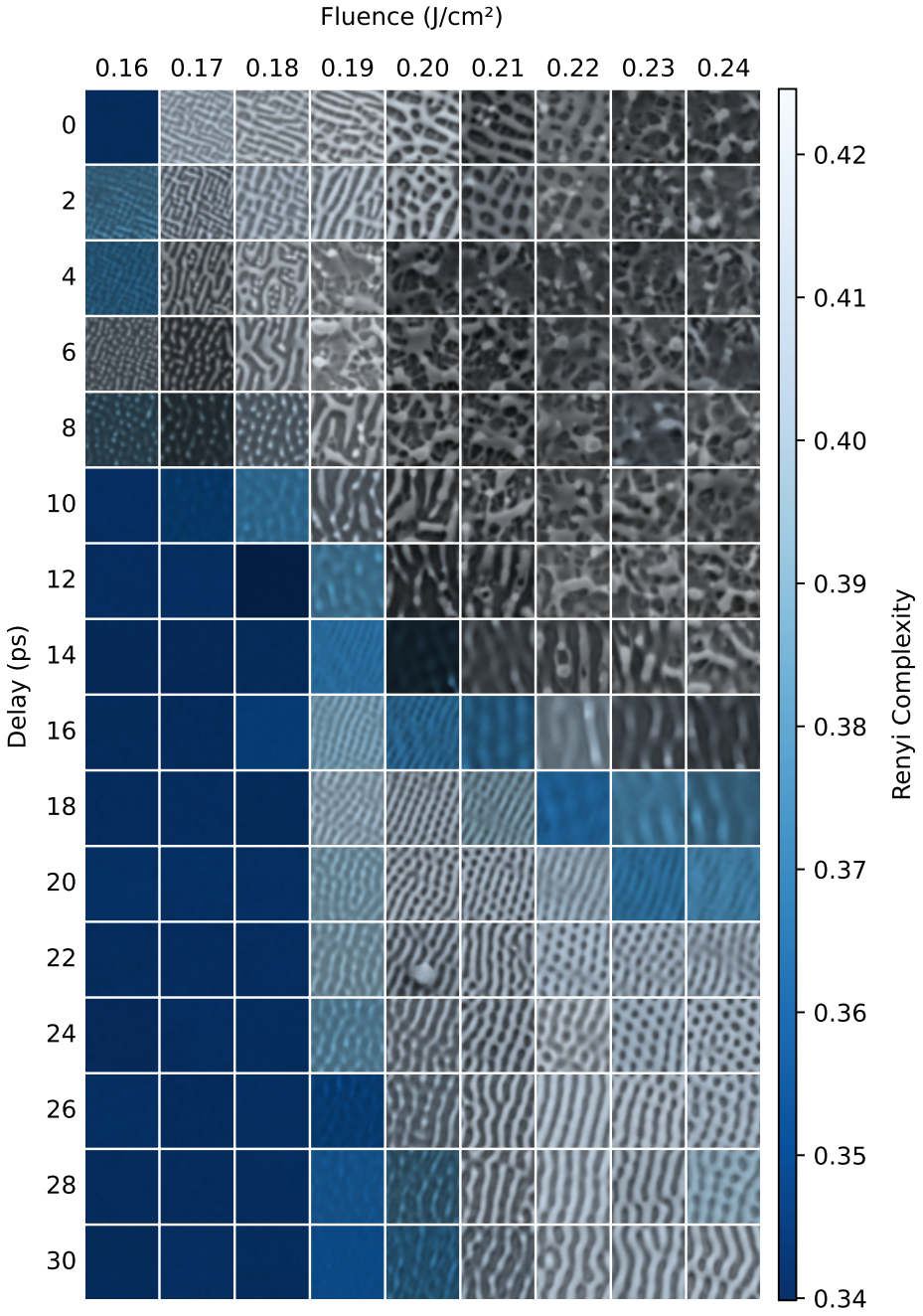


Figure A.19: Rényi Complexity of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

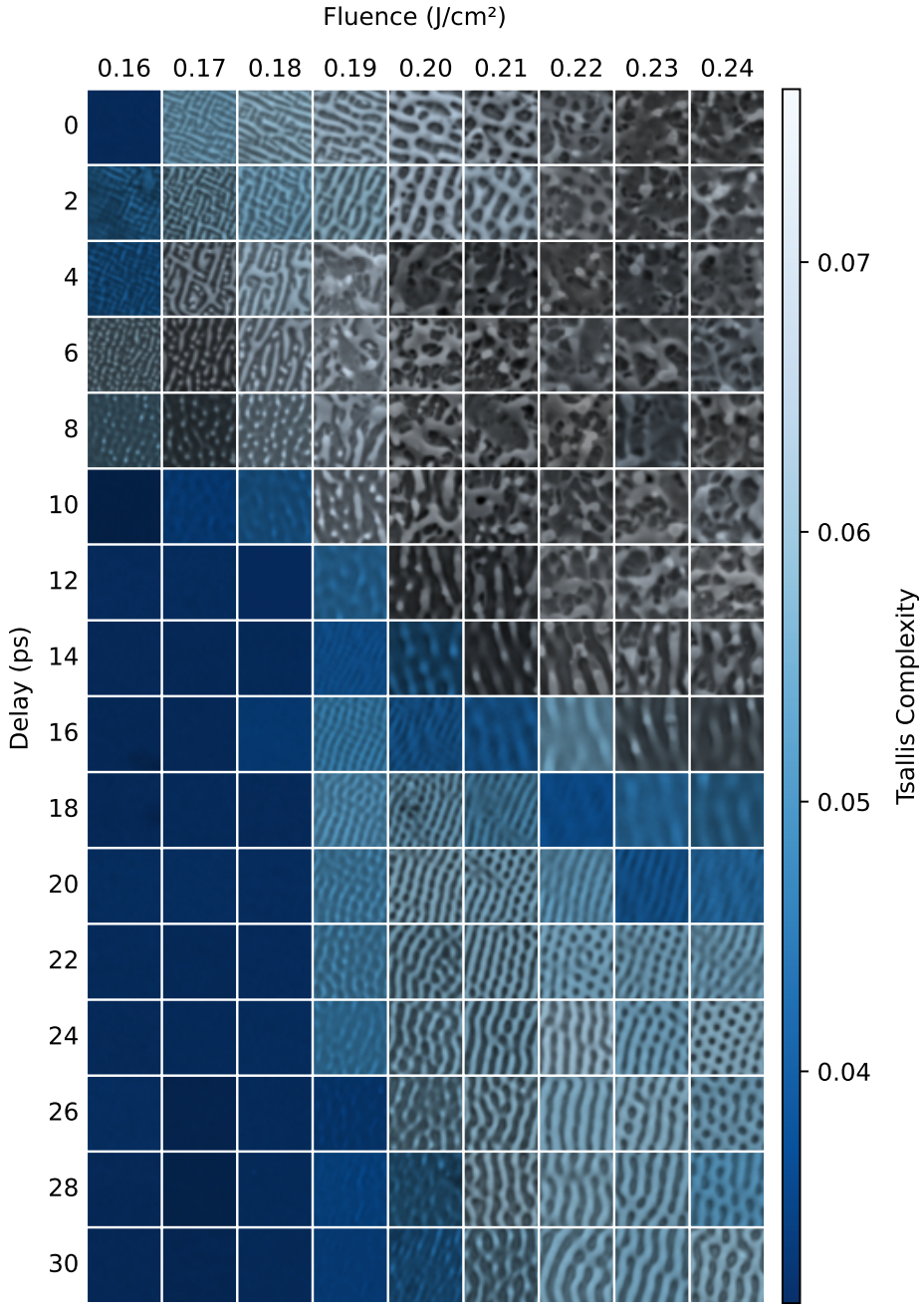


Figure A.20: Tsallis Complexity of the gray level runs (glr2) of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

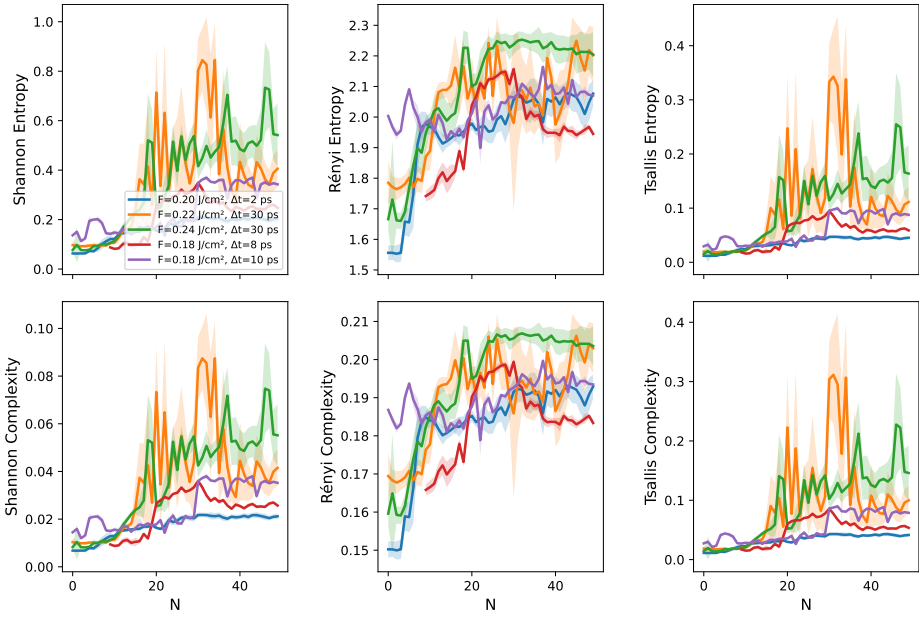


Figure A.21: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) glr1 entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

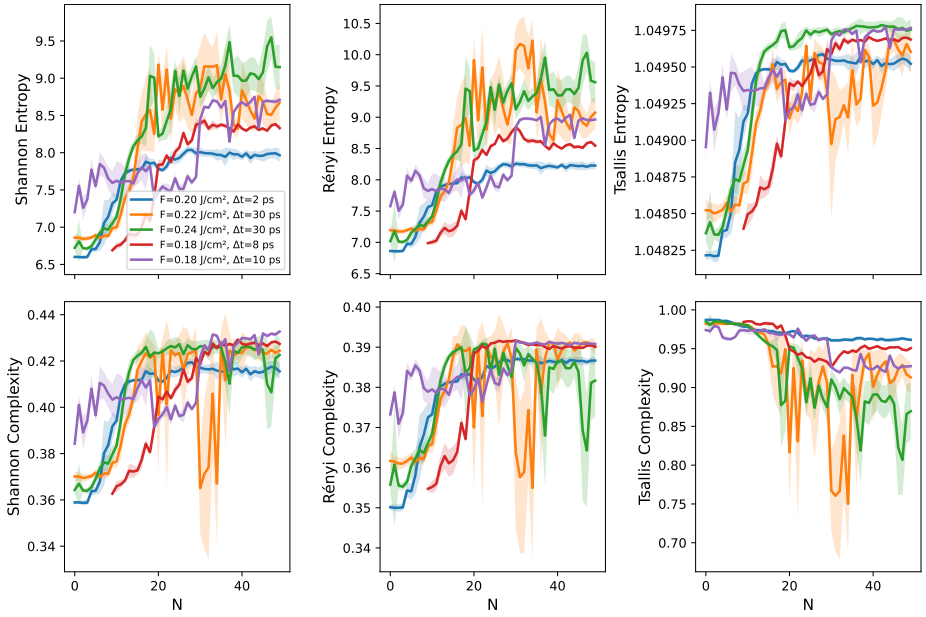


Figure A.22: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) glr2 entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

A.1.3 Fourier spectrum complexities

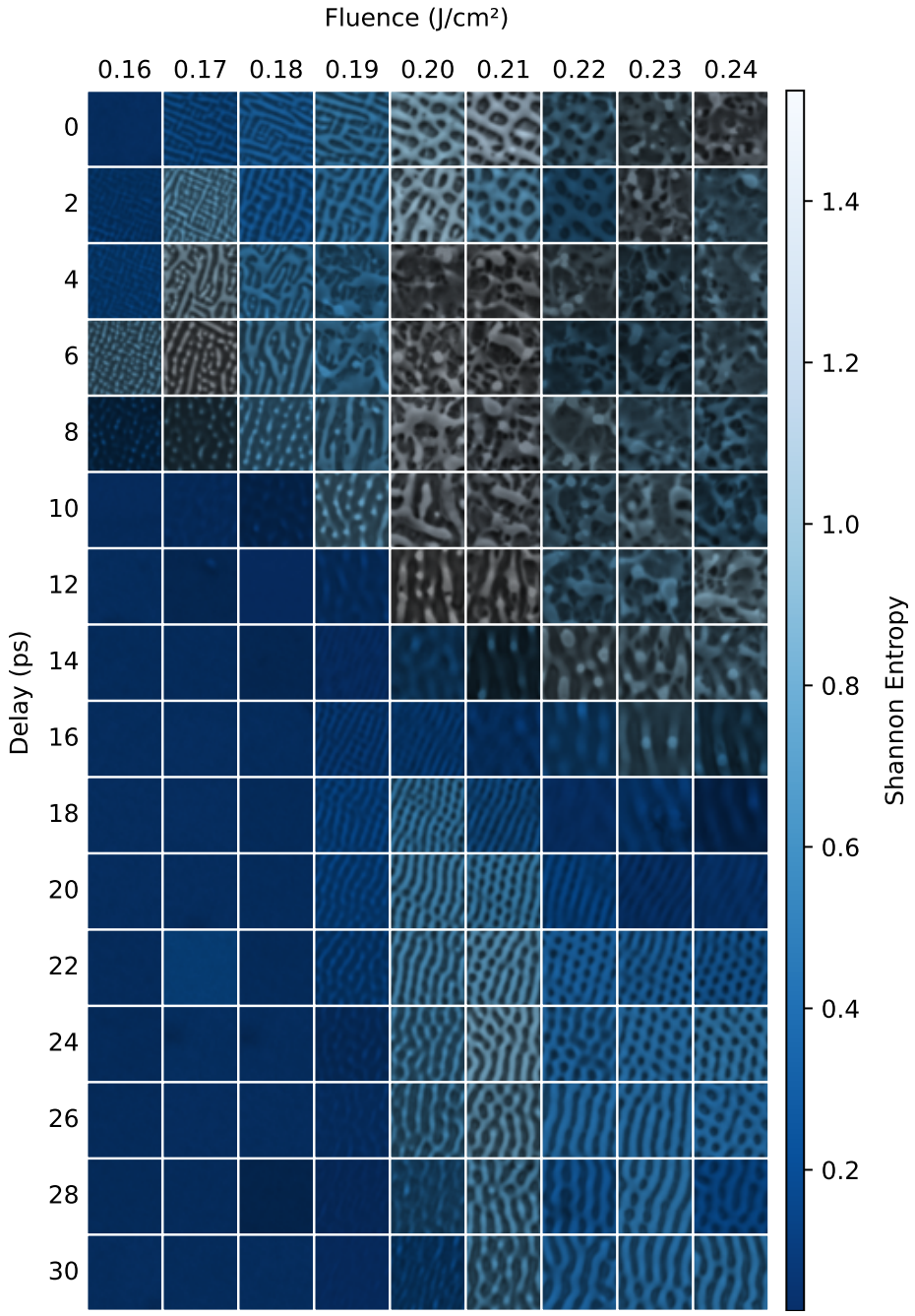


Figure A.23: Shannon entropy of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

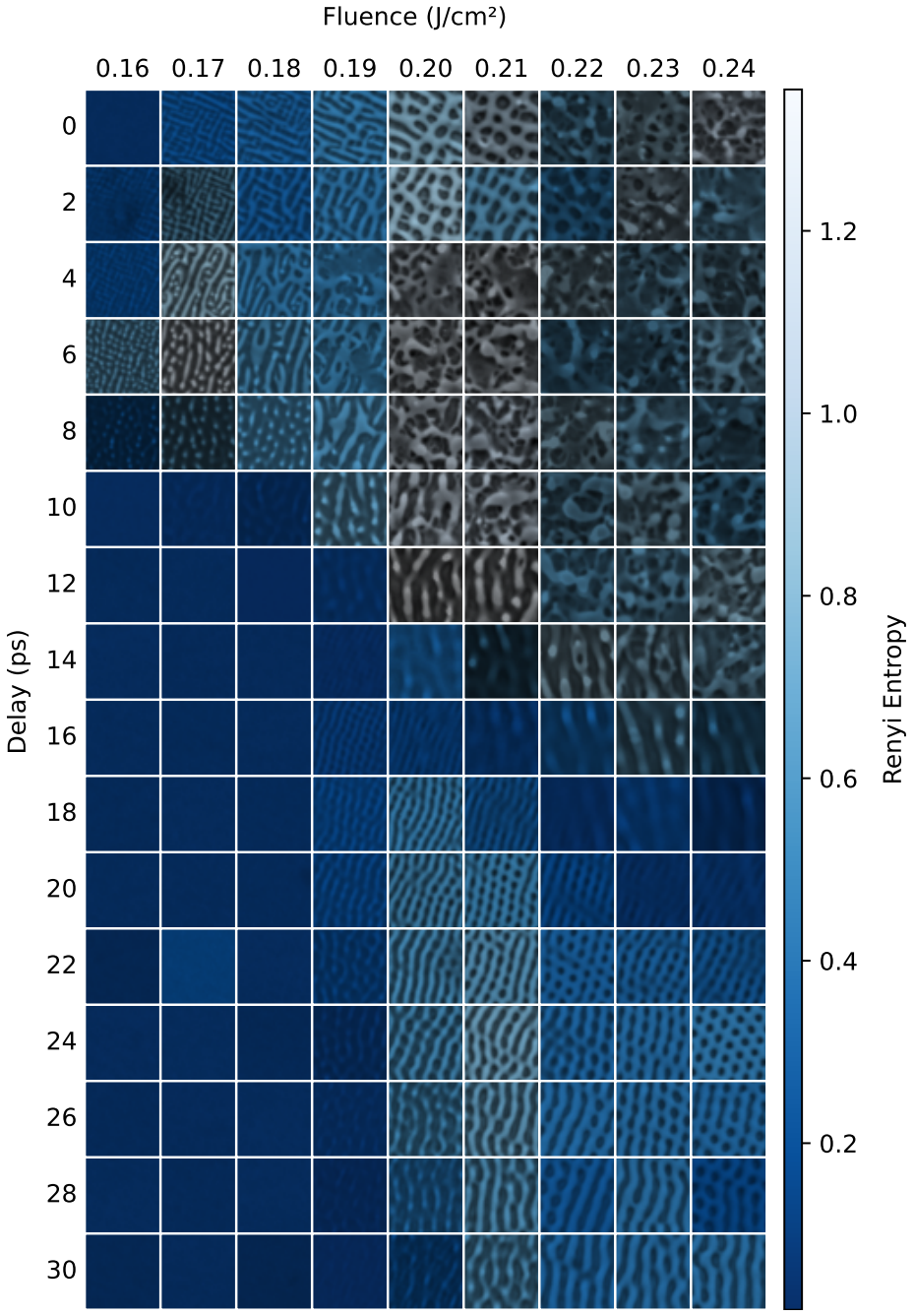


Figure A.24: Rényi entropy of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

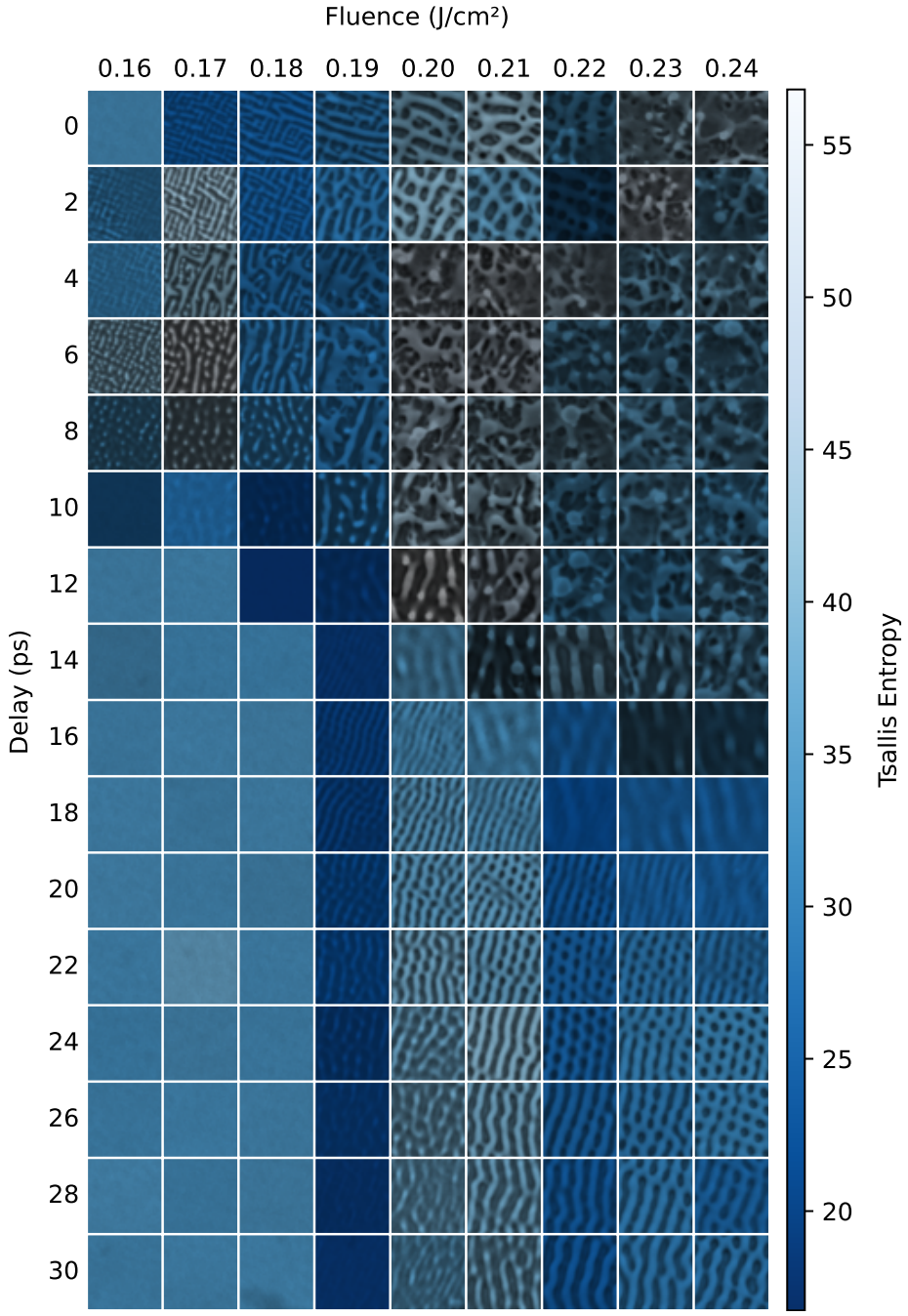


Figure A.25: Tsallis entropy of of the Fourier power spectrum SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

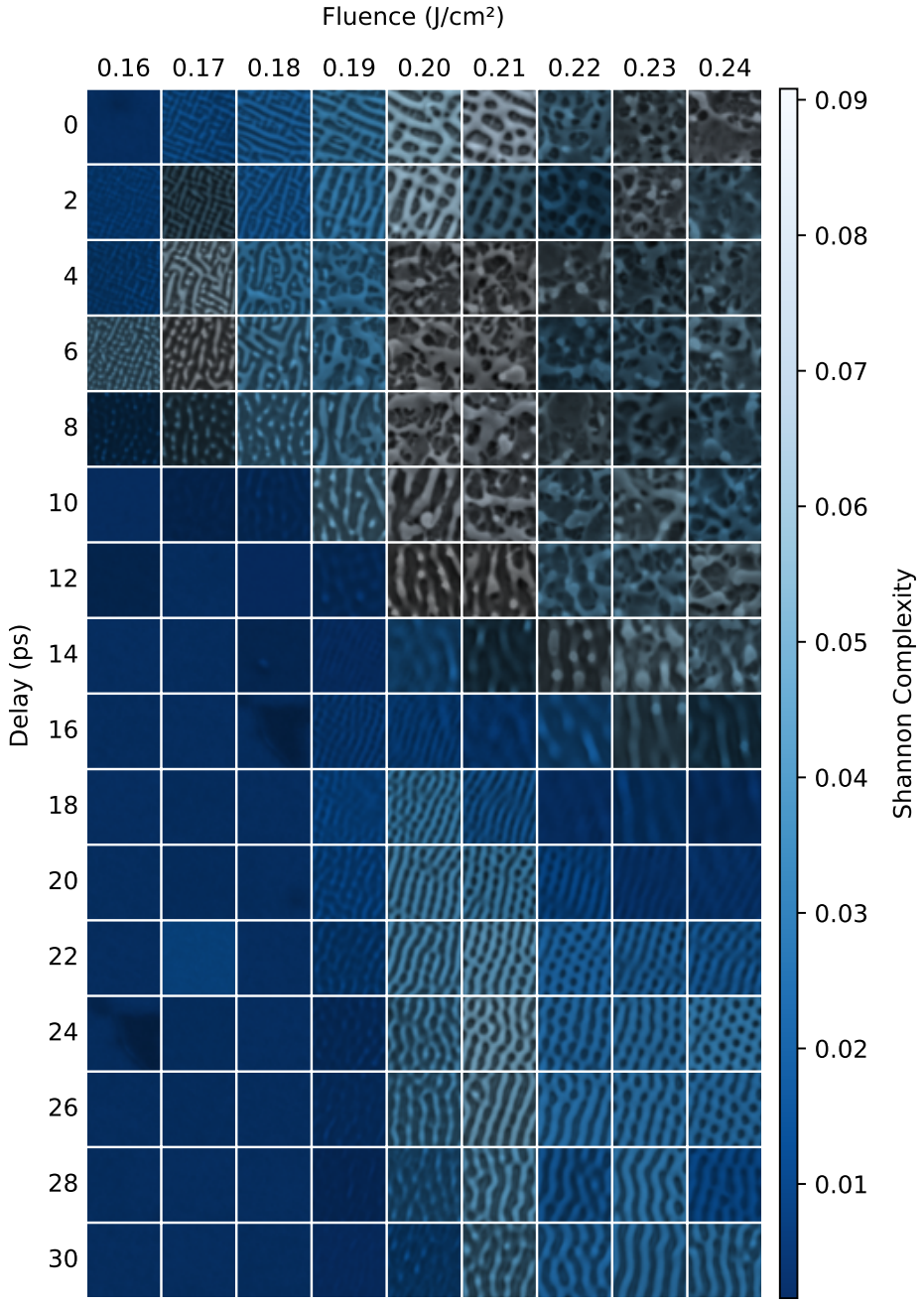


Figure A.26: Shannon complexity of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

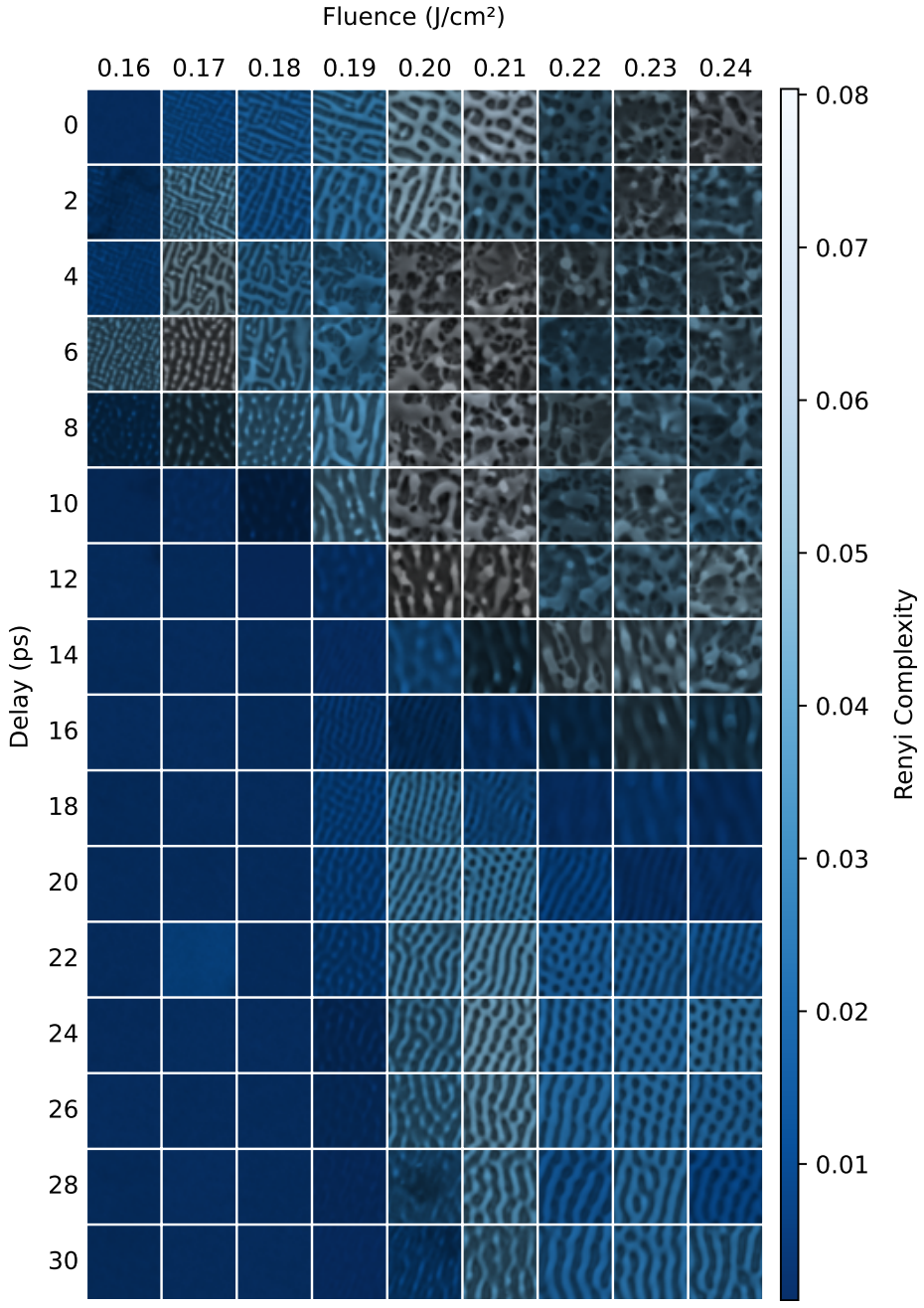


Figure A.27: Rényi Complexity of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

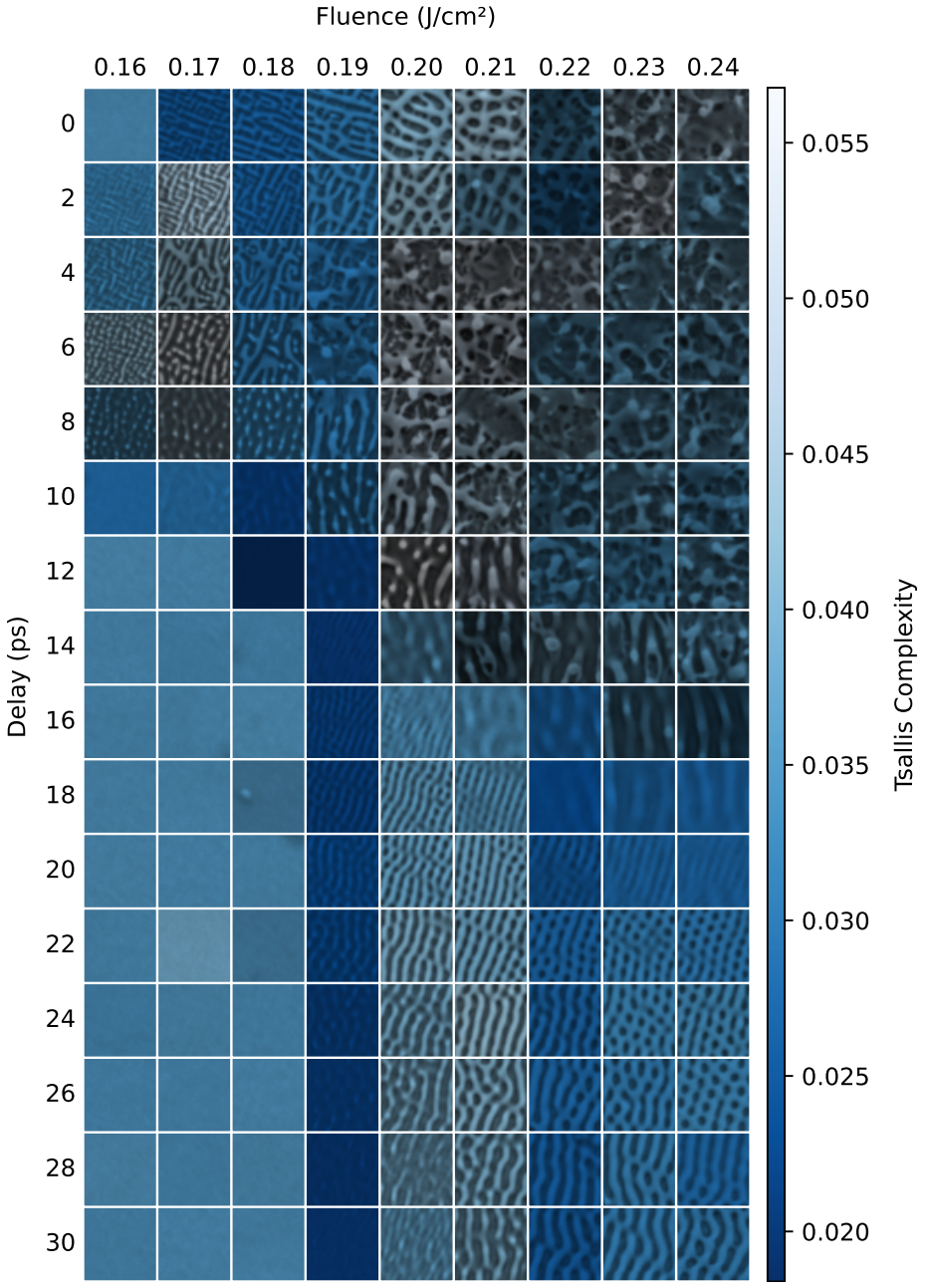


Figure A.28: Tsallis Complexity of the Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

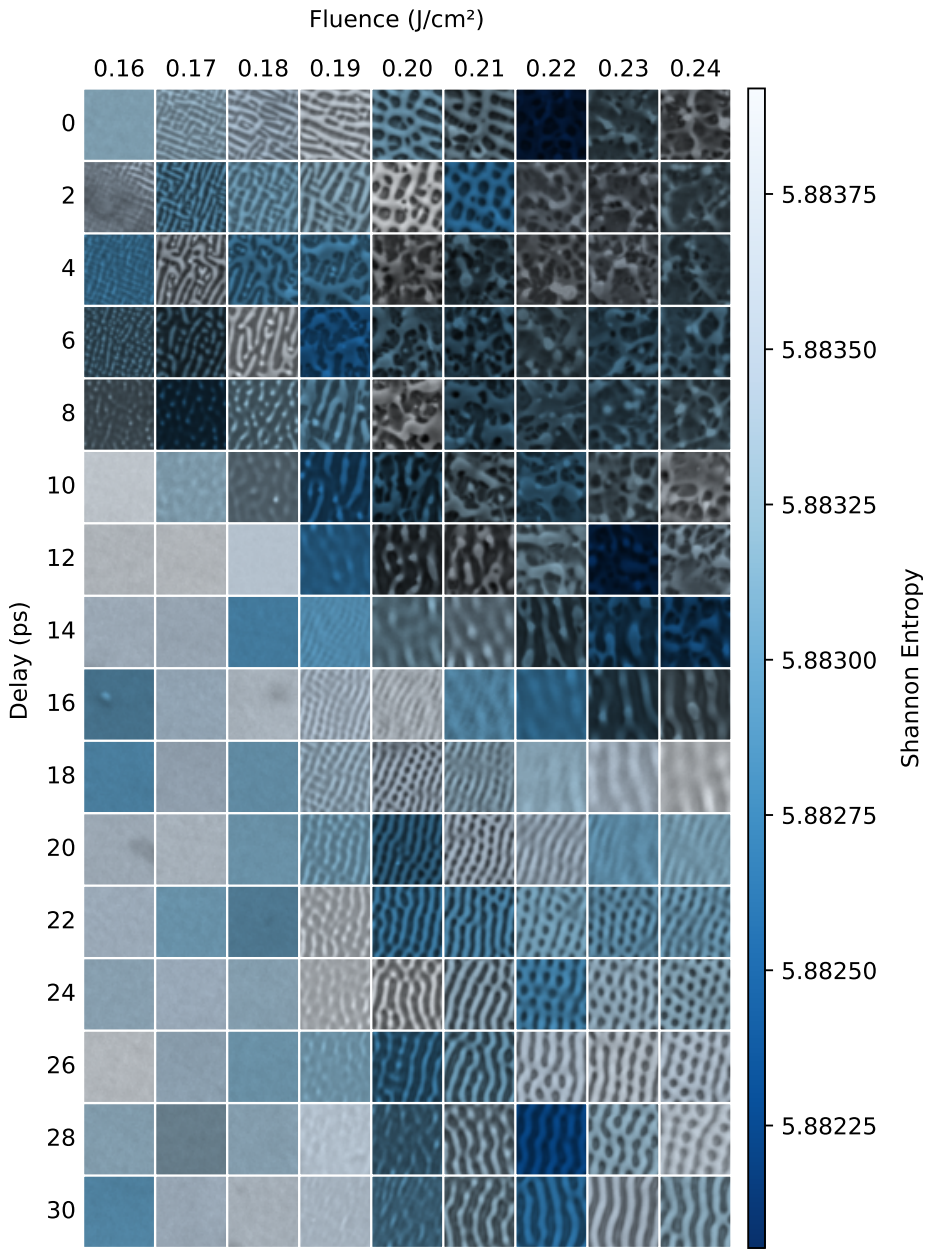


Figure A.29: Shannon entropy of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

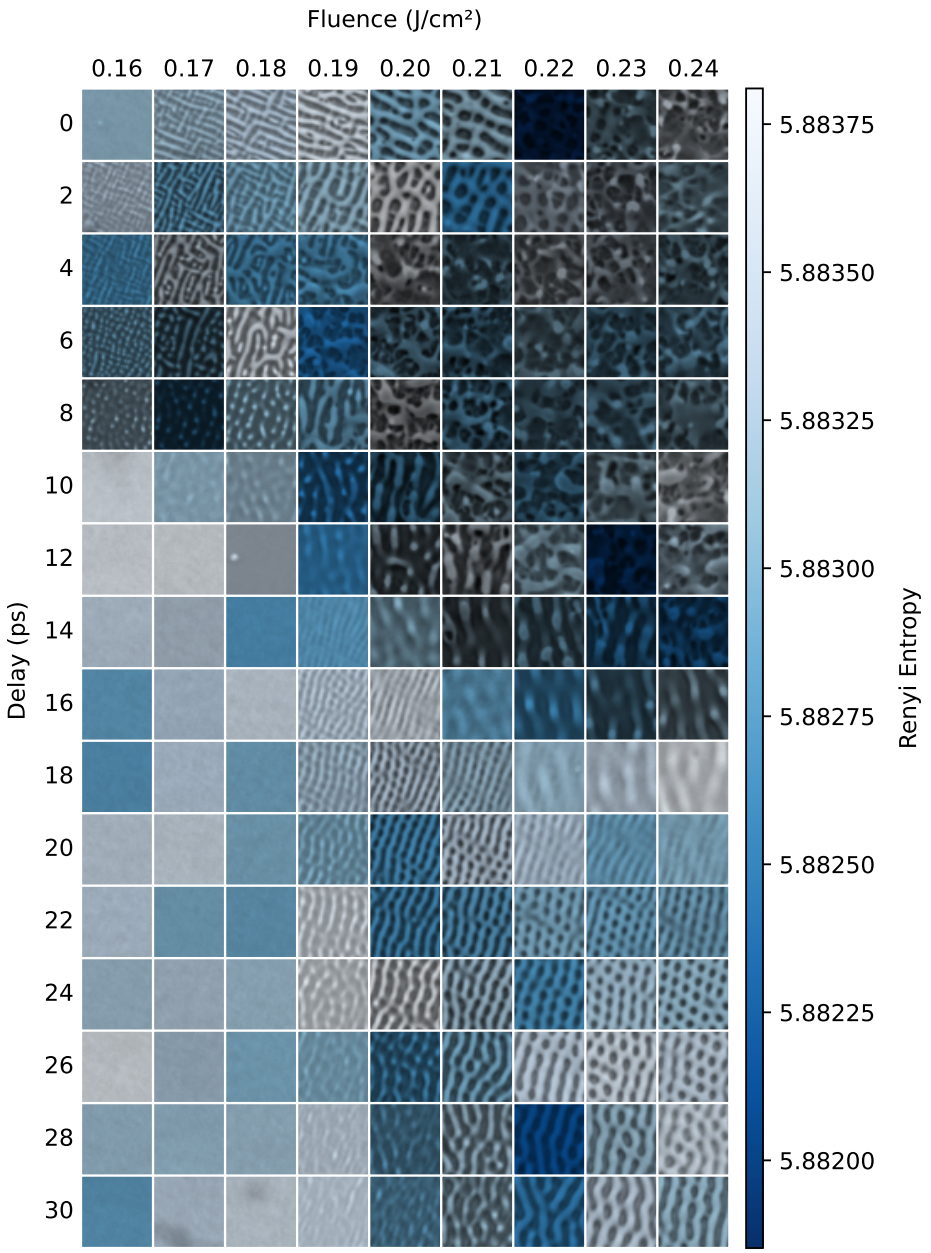


Figure A.30: Rényi entropy of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

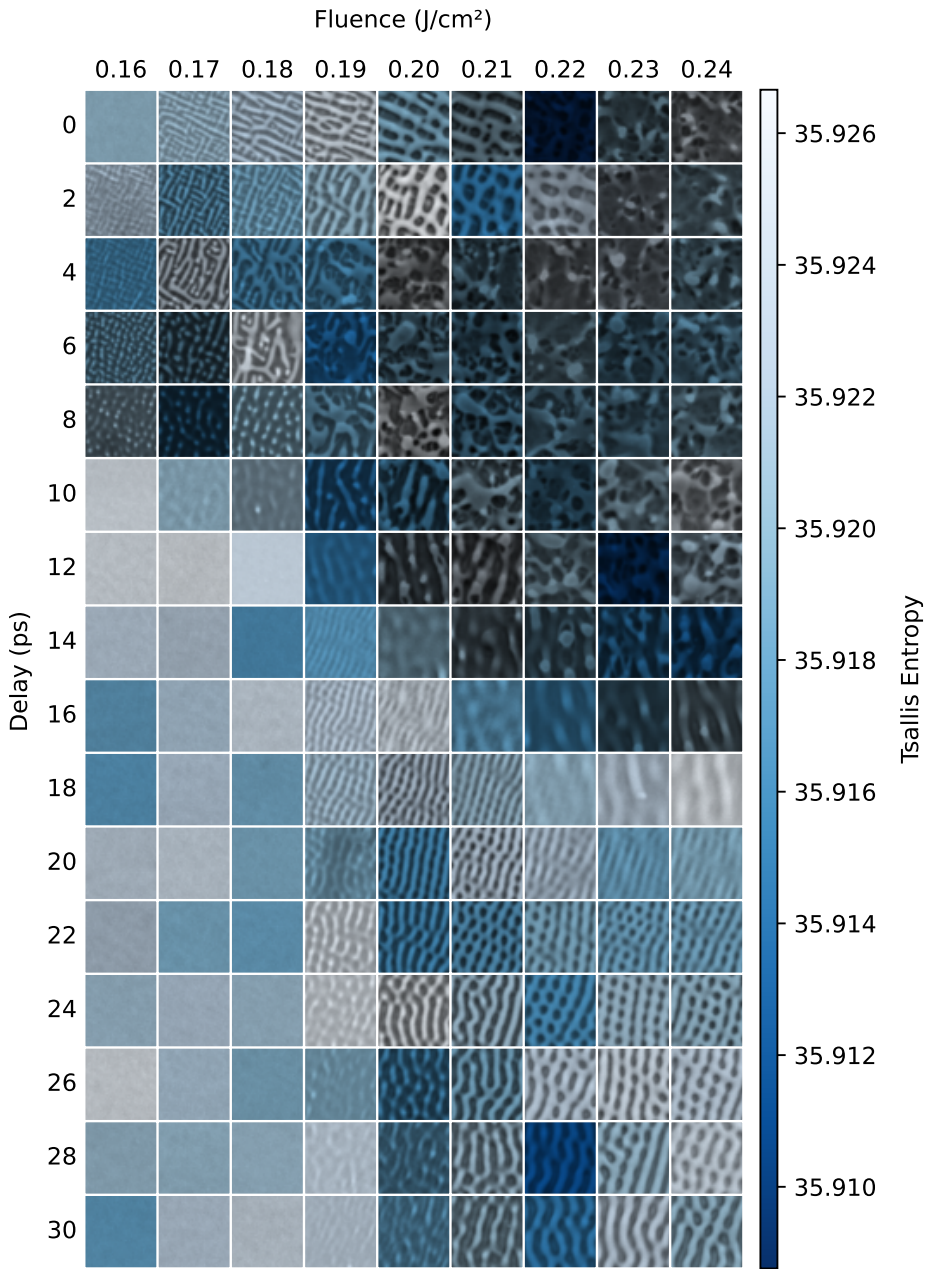


Figure A.31: Tsallis entropy of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

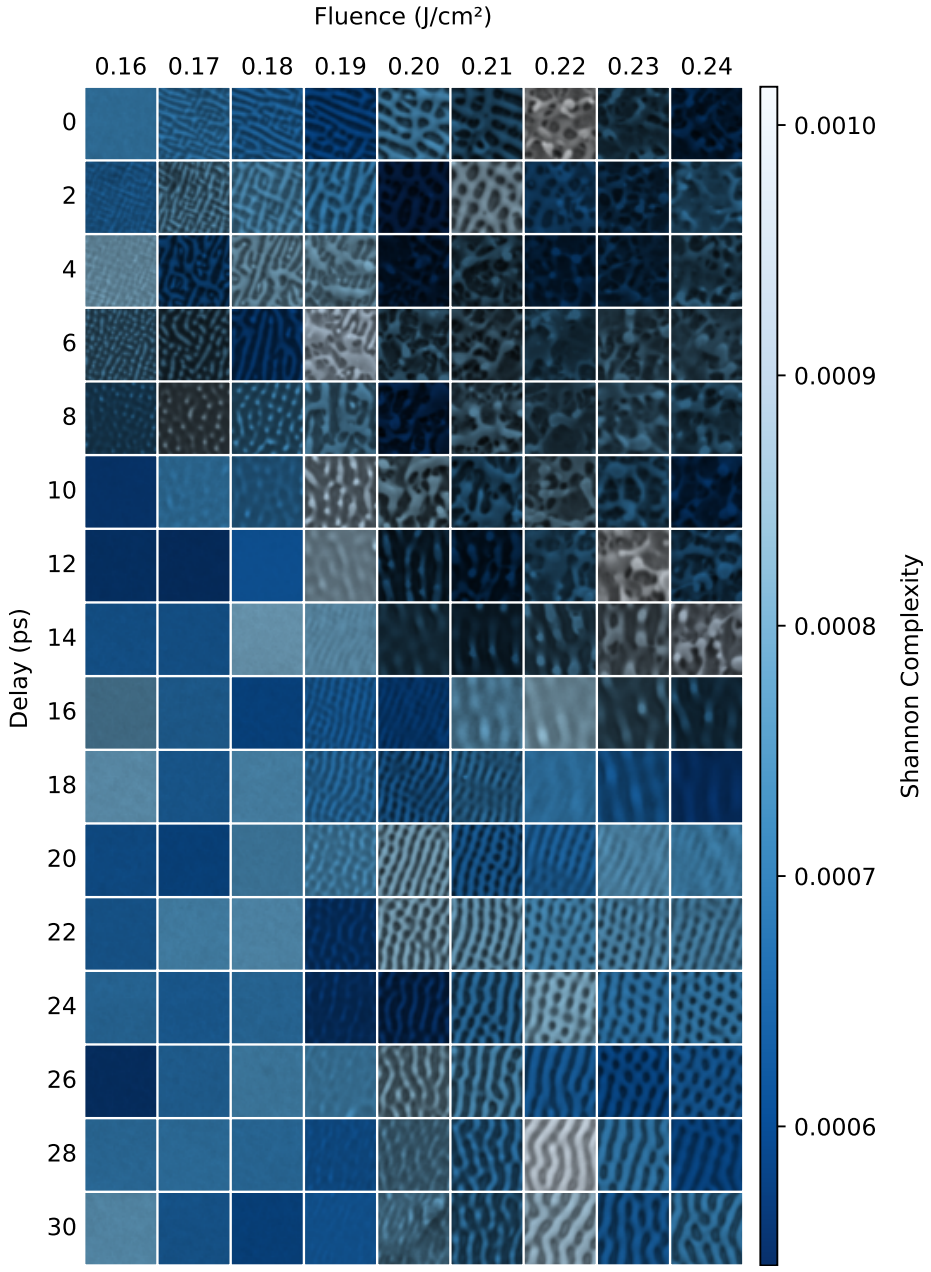


Figure A.32: Shannon complexity of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

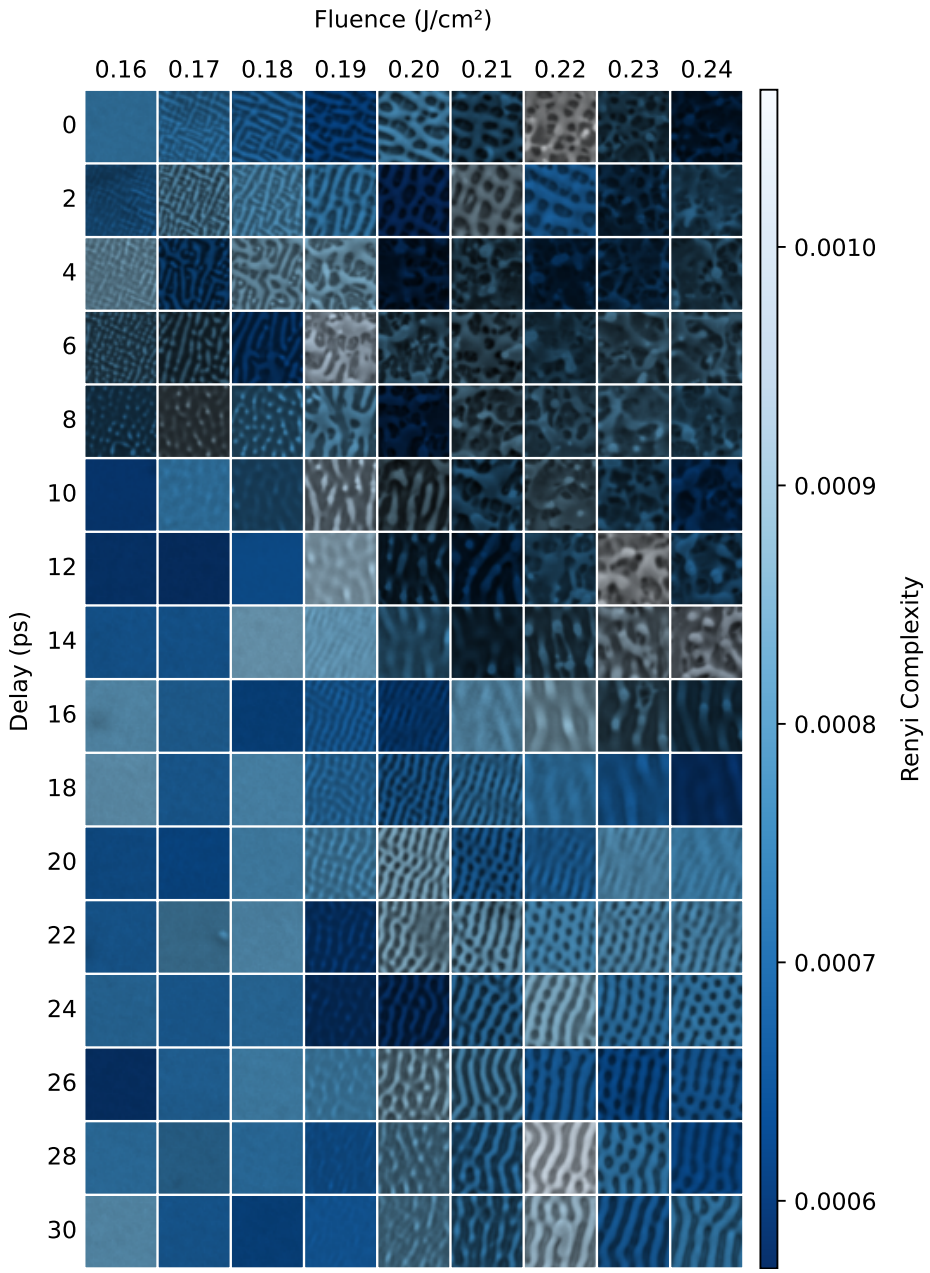


Figure A.33: Rényi Complexity of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

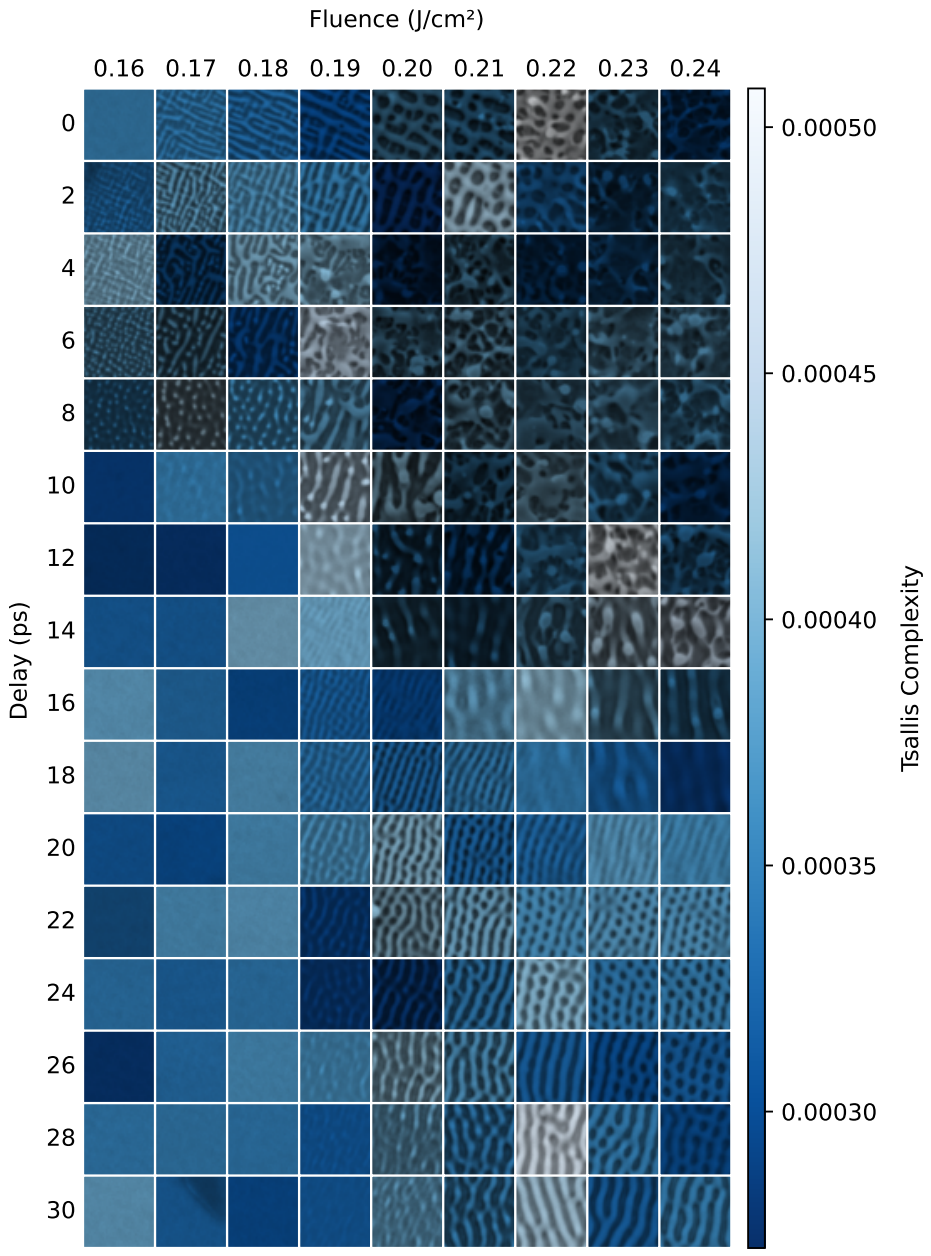


Figure A.34: Tsallis Complexity of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

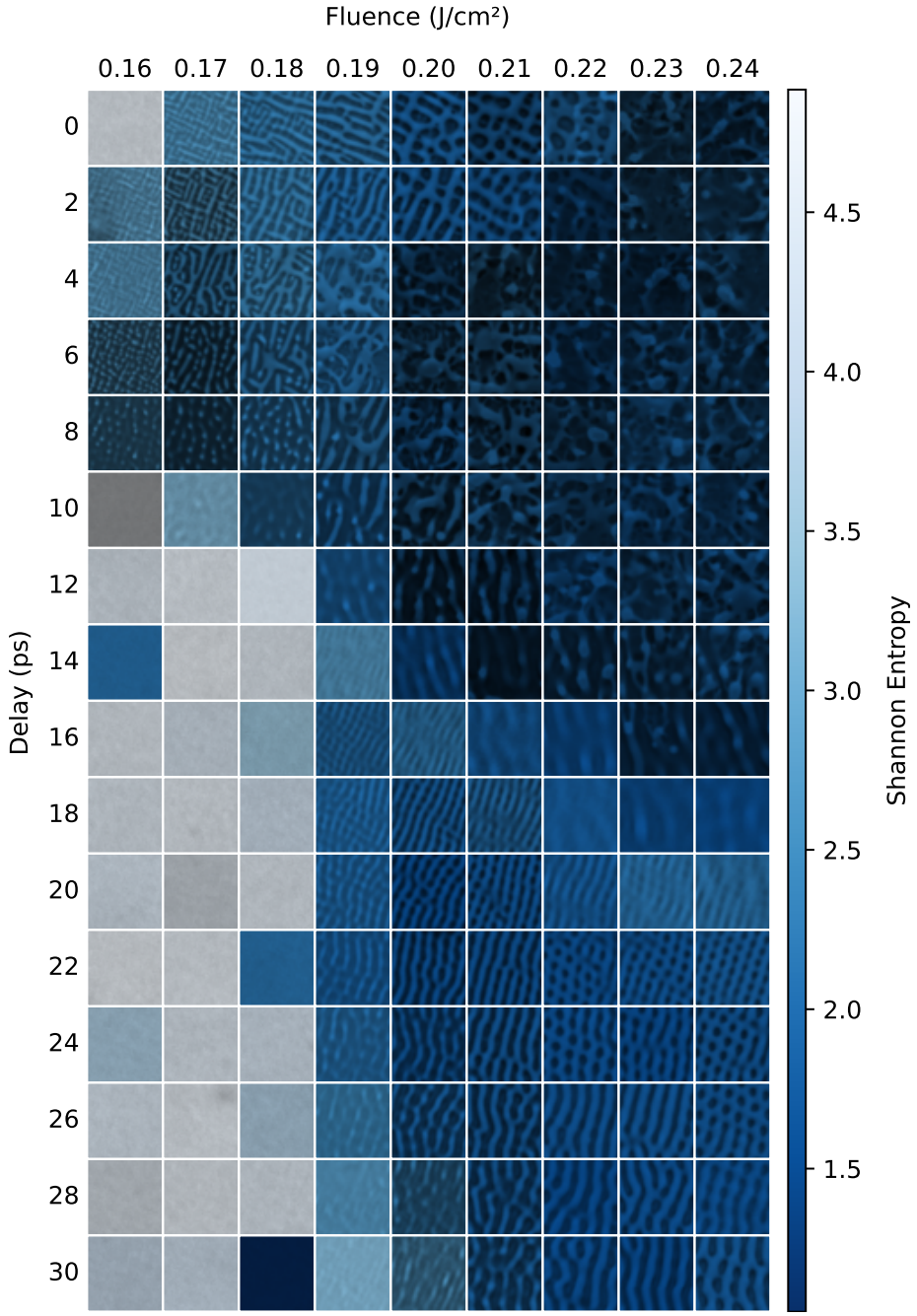


Figure A.35: Shannon entropy of the azimuthally averaged Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

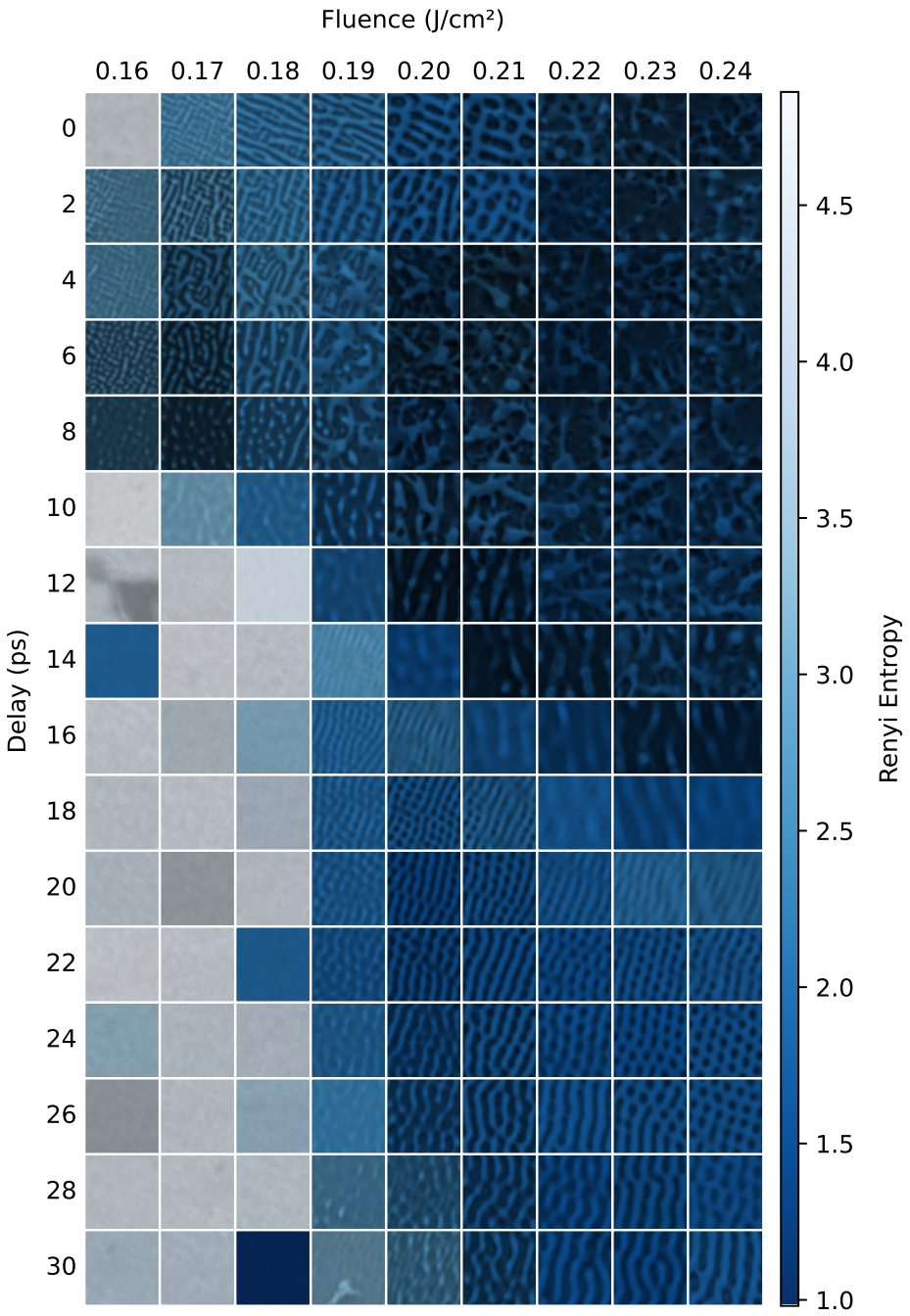


Figure A.36: Rényi entropy of the azimuthally averaged Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

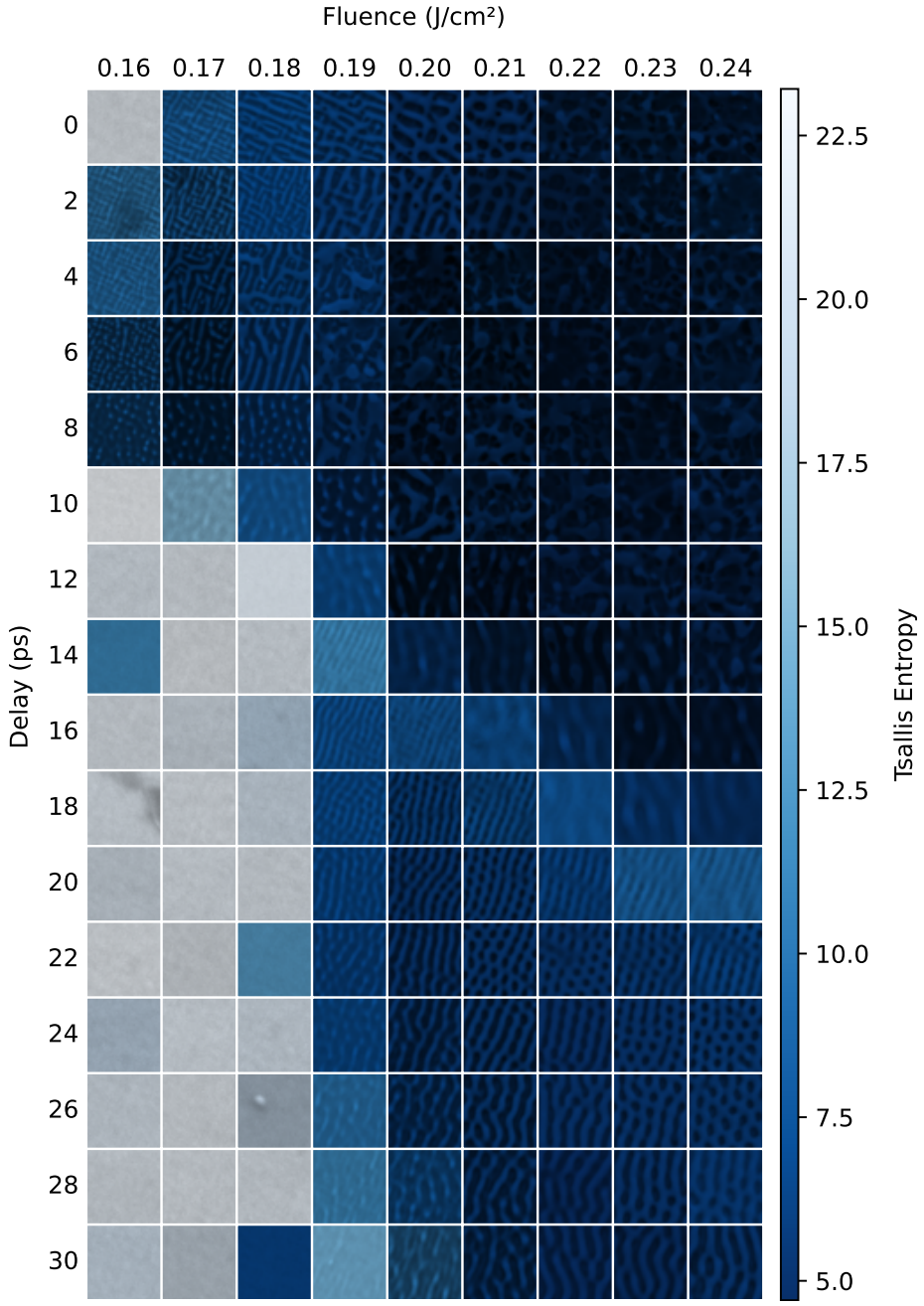


Figure A.37: Tsallis entropy of the Fourier phase spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

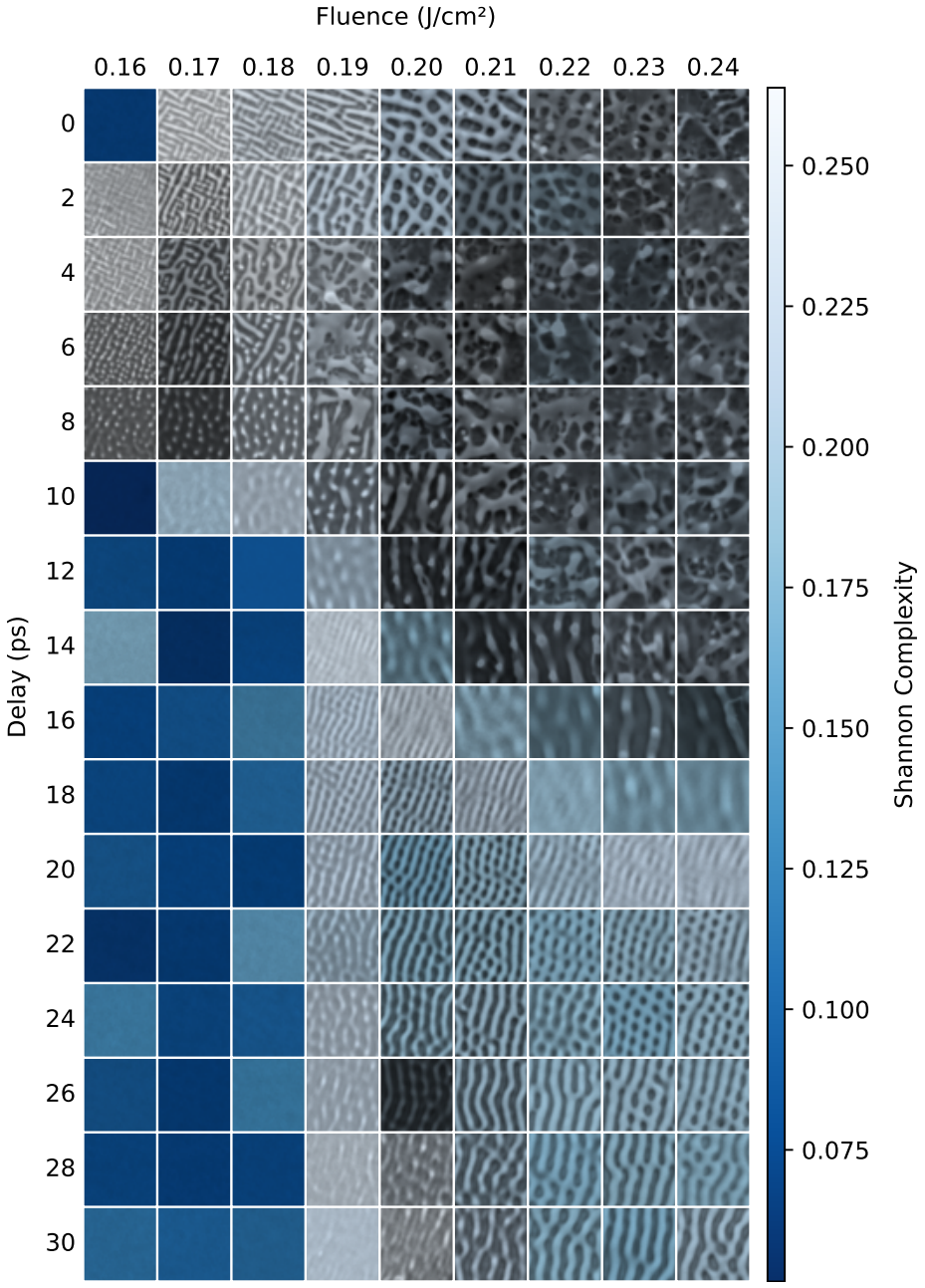


Figure A.38: Shannon complexity of the azimuthally averaged Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

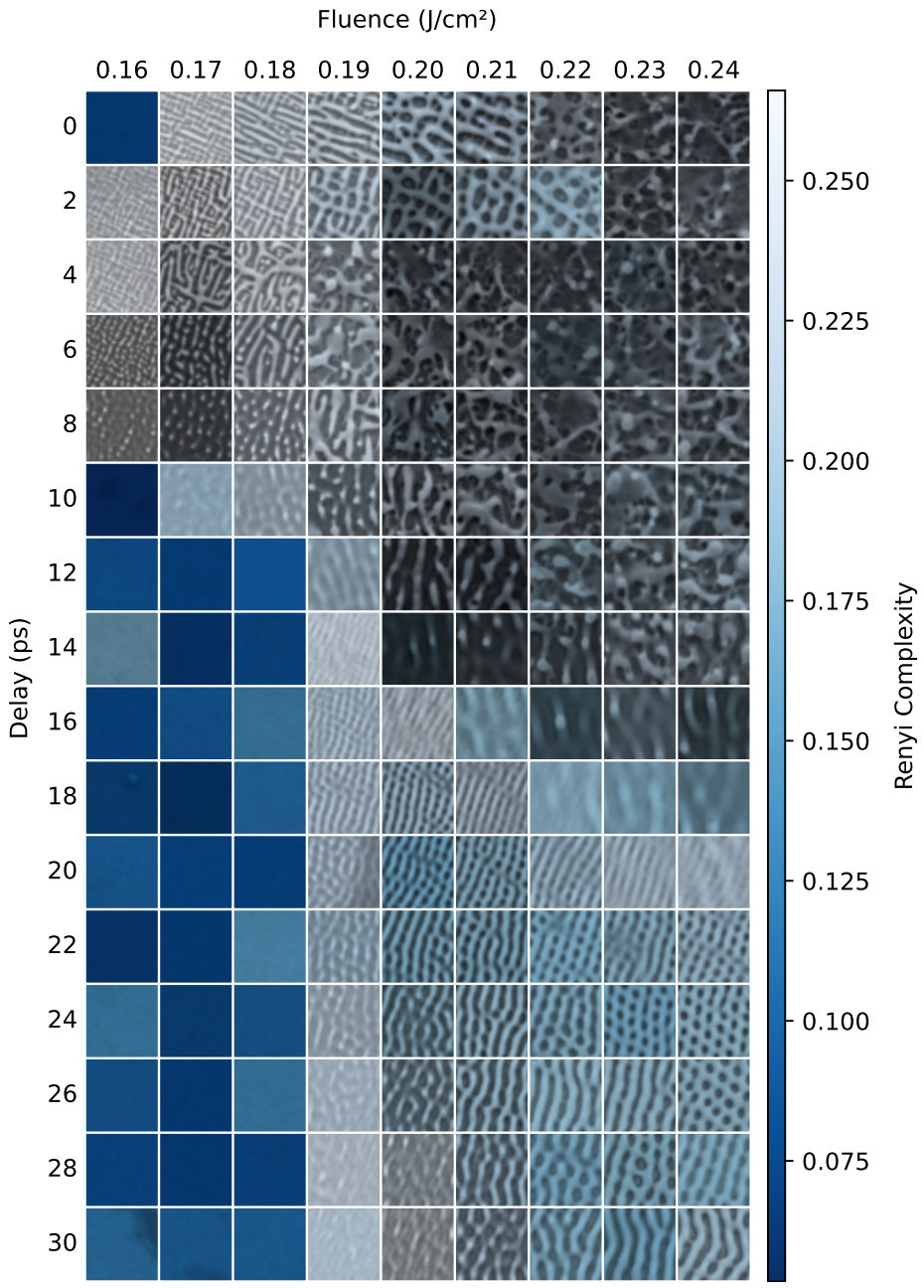


Figure A.39: Rényi Complexity of the azimuthally averaged Fourier power spectrum of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

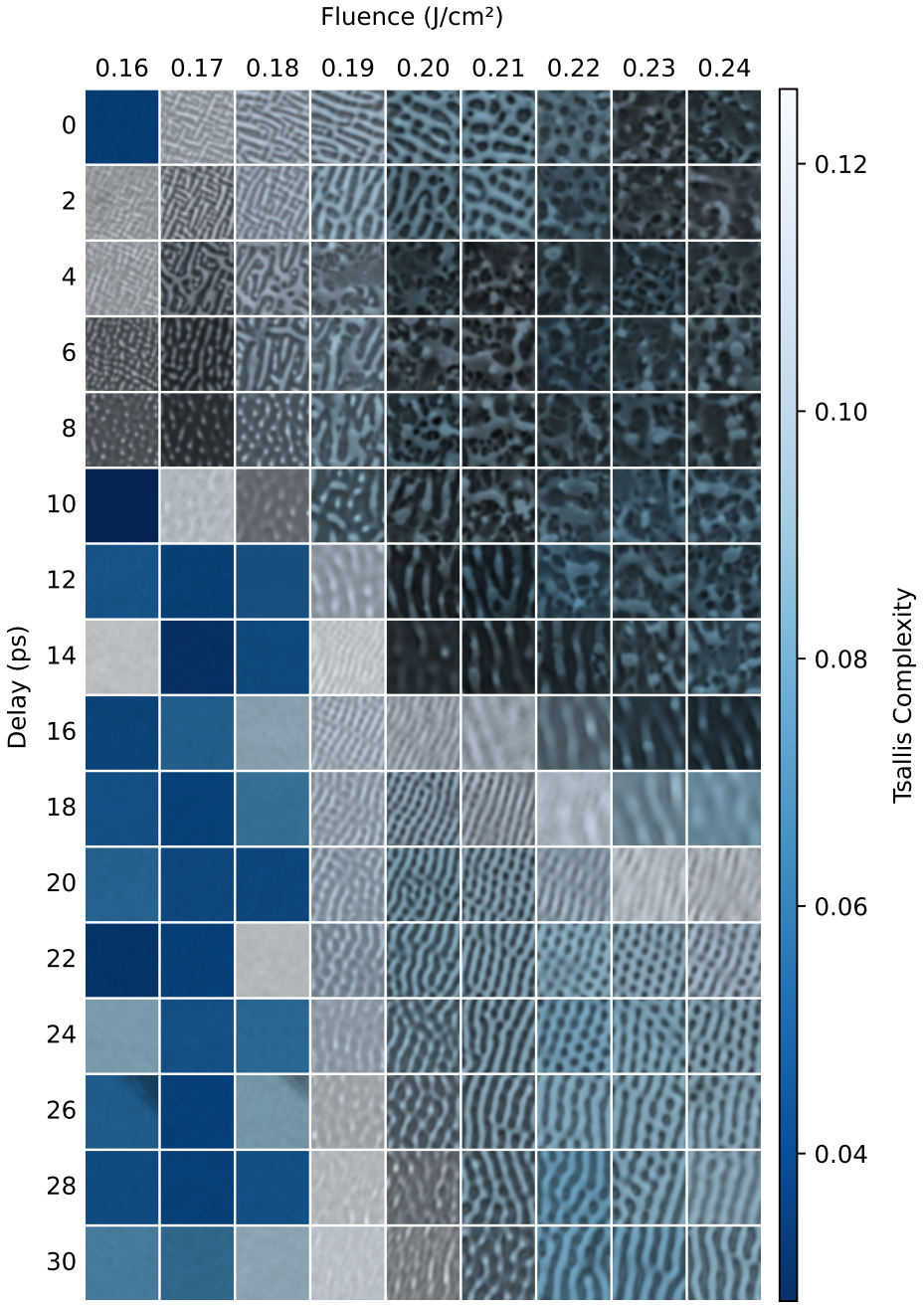


Figure A.40: Tsallis Complexity of the azimuthally averaged Fourier power spectrum SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

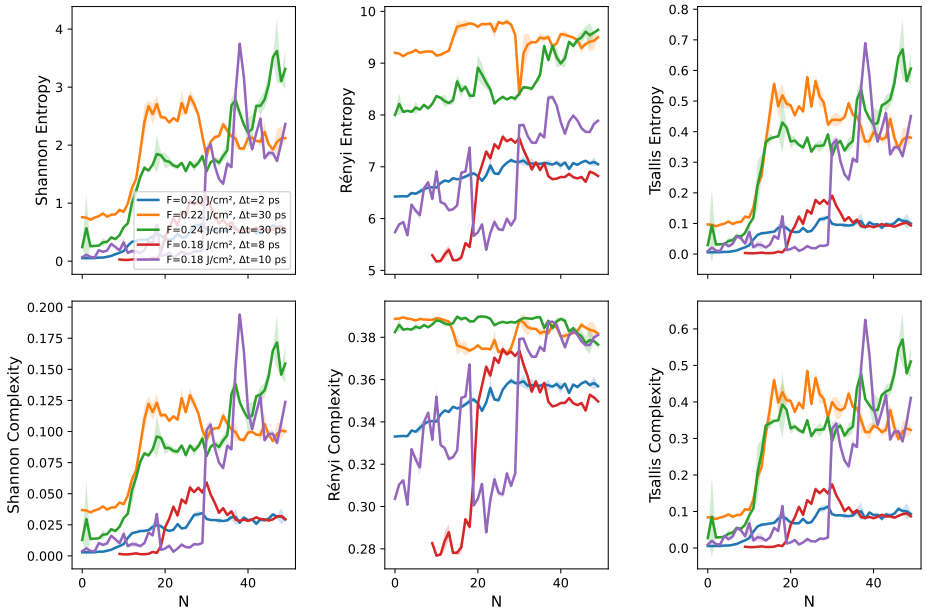


Figure A.41: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) power spectral entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

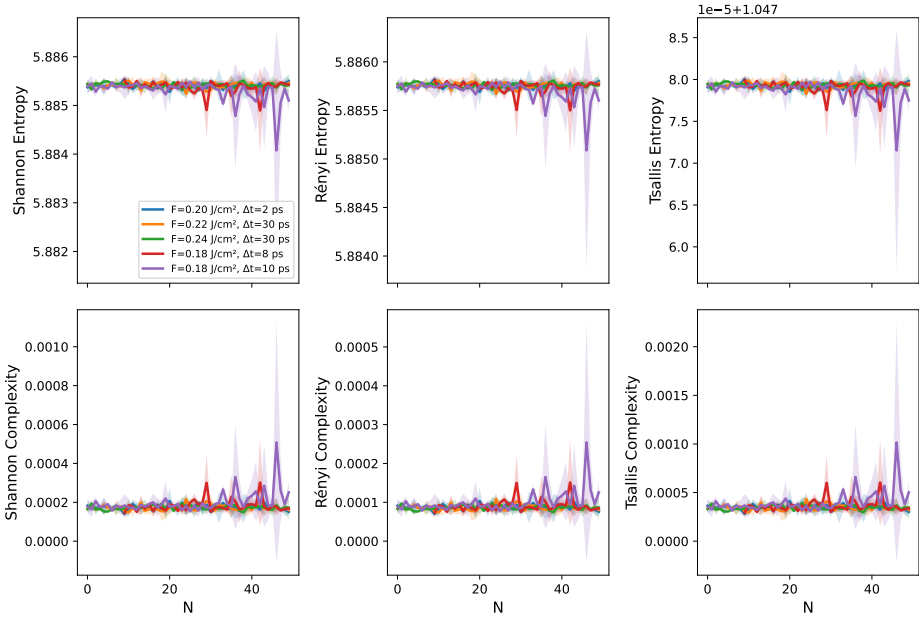


Figure A.42: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) phase spectrum entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

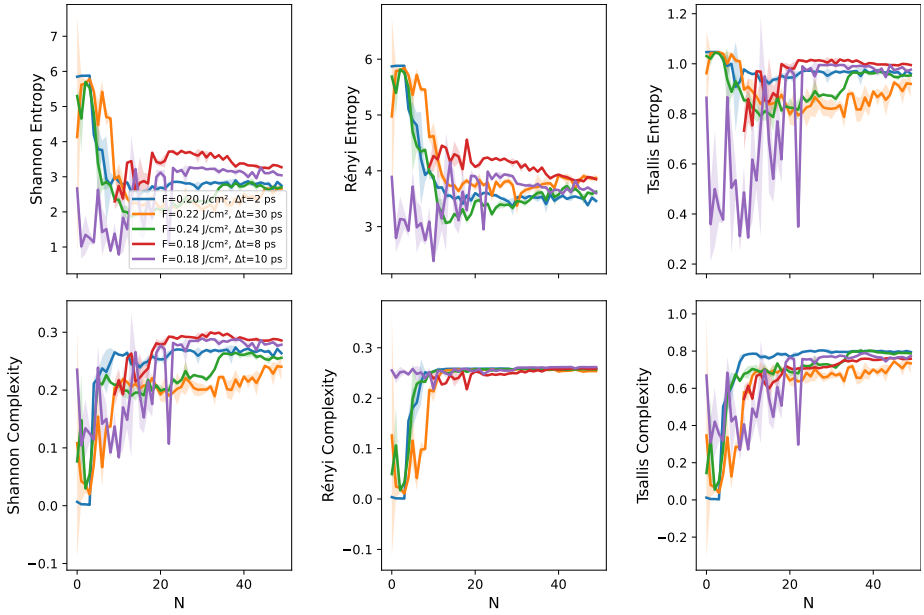


Figure A.43: Comparison of Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) of azimuthally averaged power spectrum entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

A.1.4 Lempel-Ziv complexities

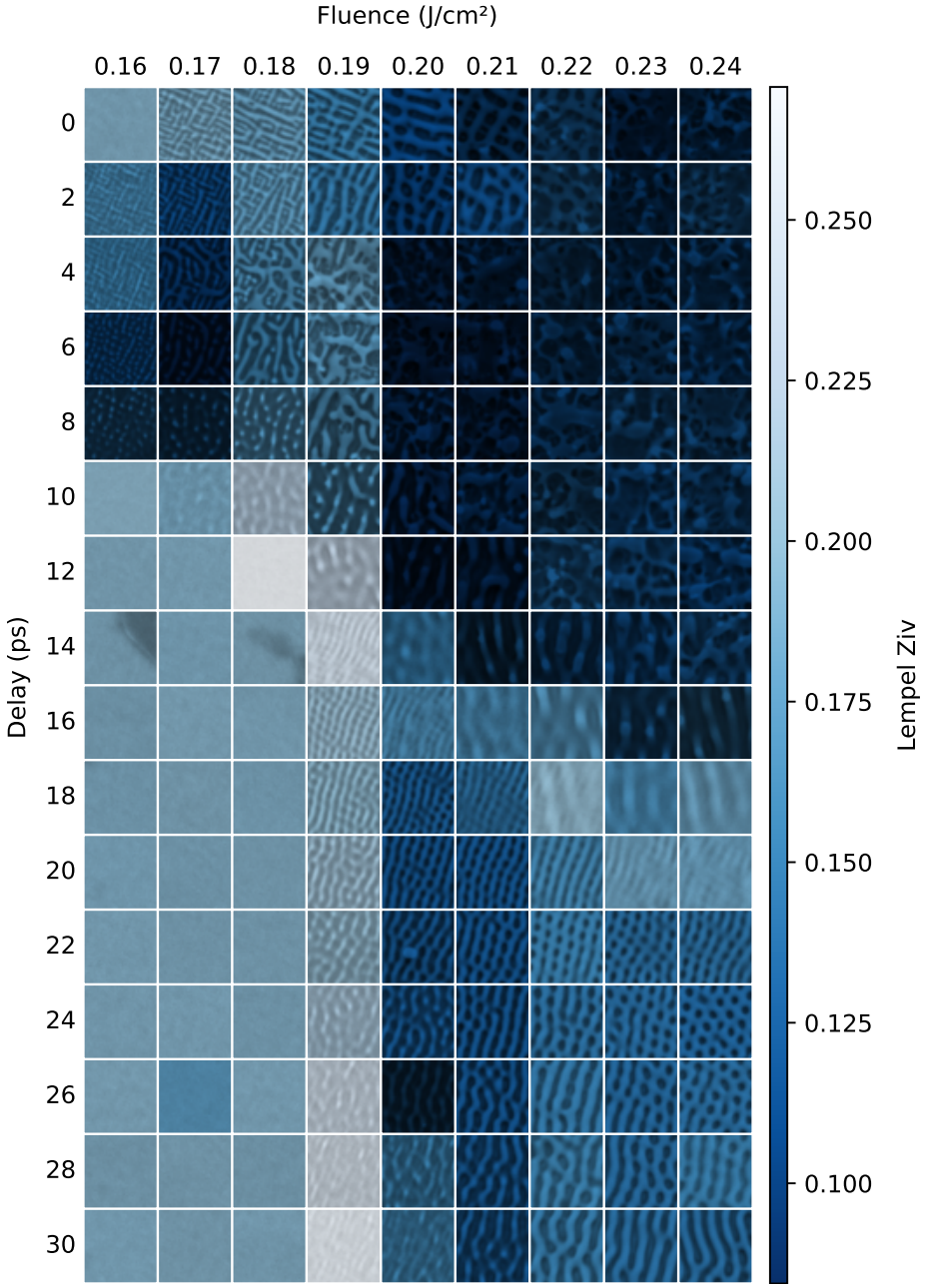


Figure A.44: Lempel-Ziv complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

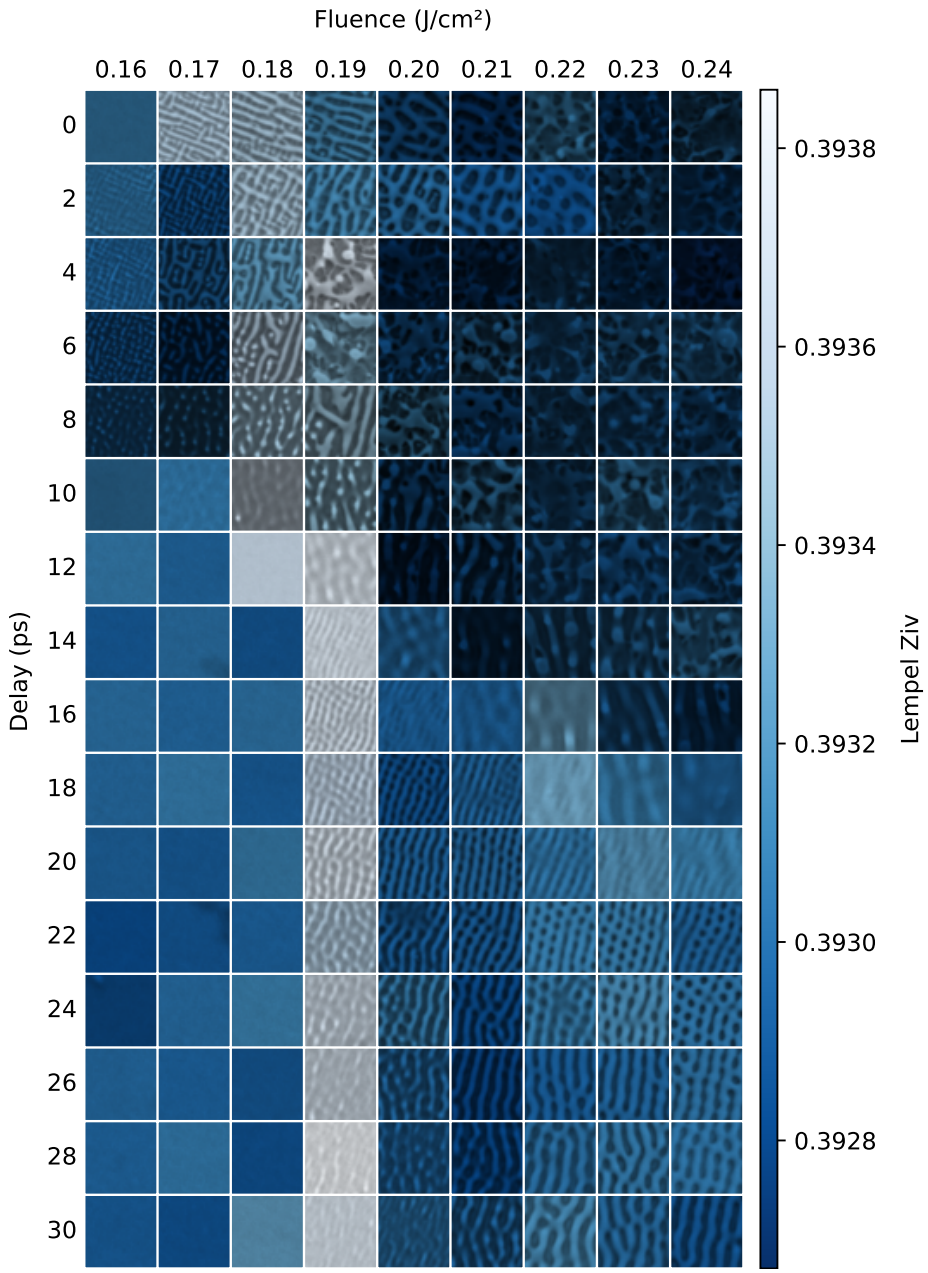


Figure A.45: Lempel-Ziv complexity of the Power spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

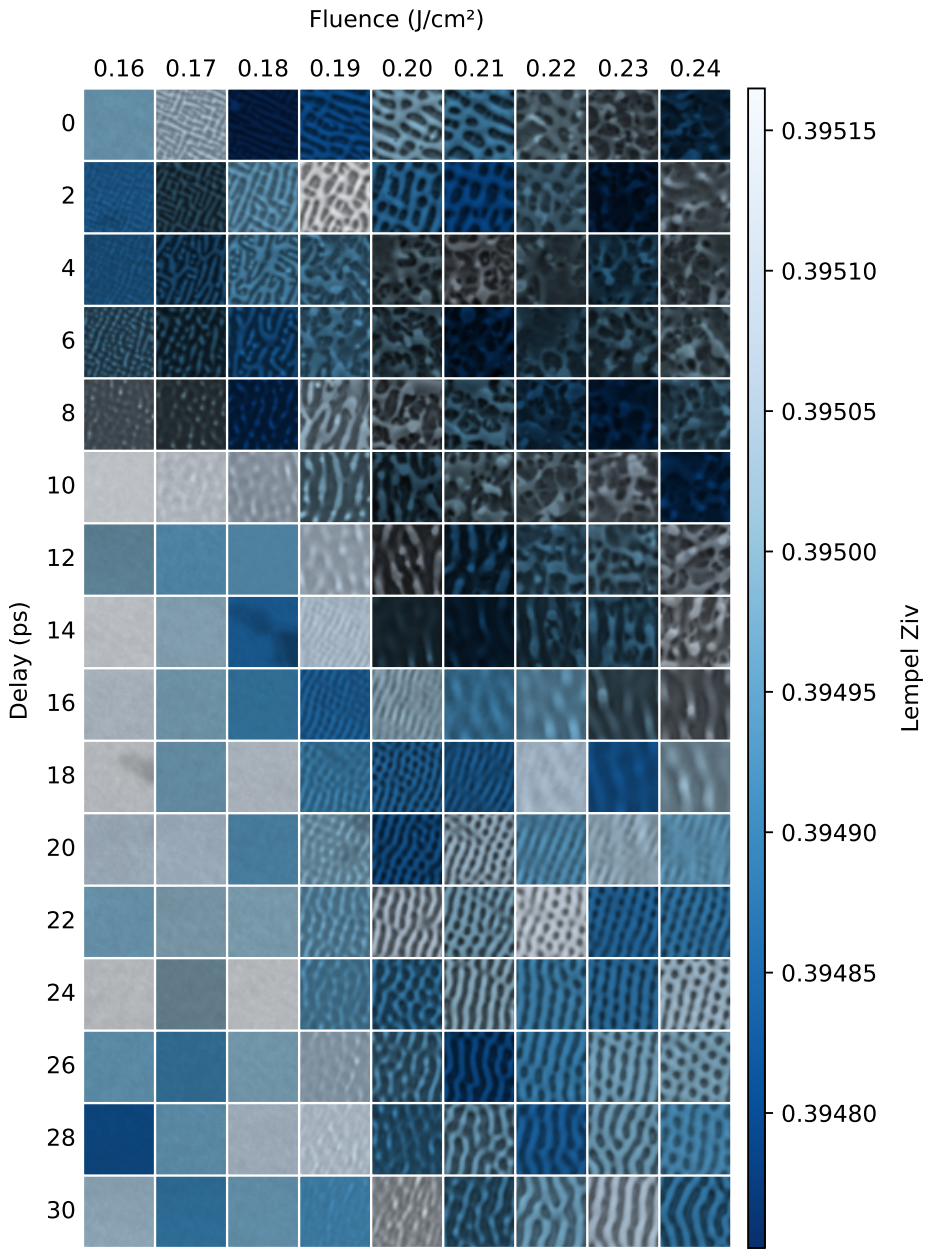


Figure A.46: Lempel-Ziv complexity of the Phase spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

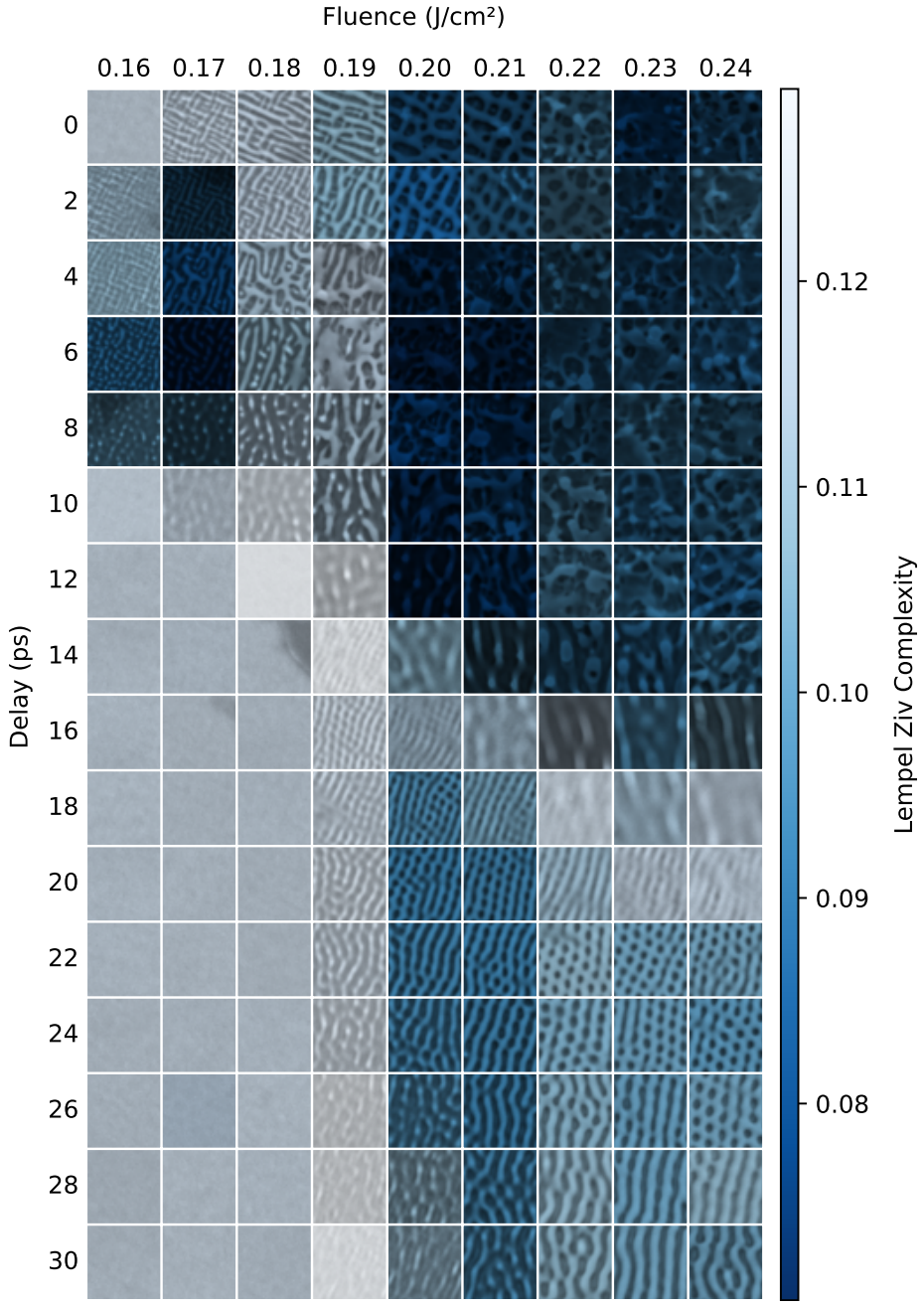


Figure A.47: Intensive Lempel-Ziv complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

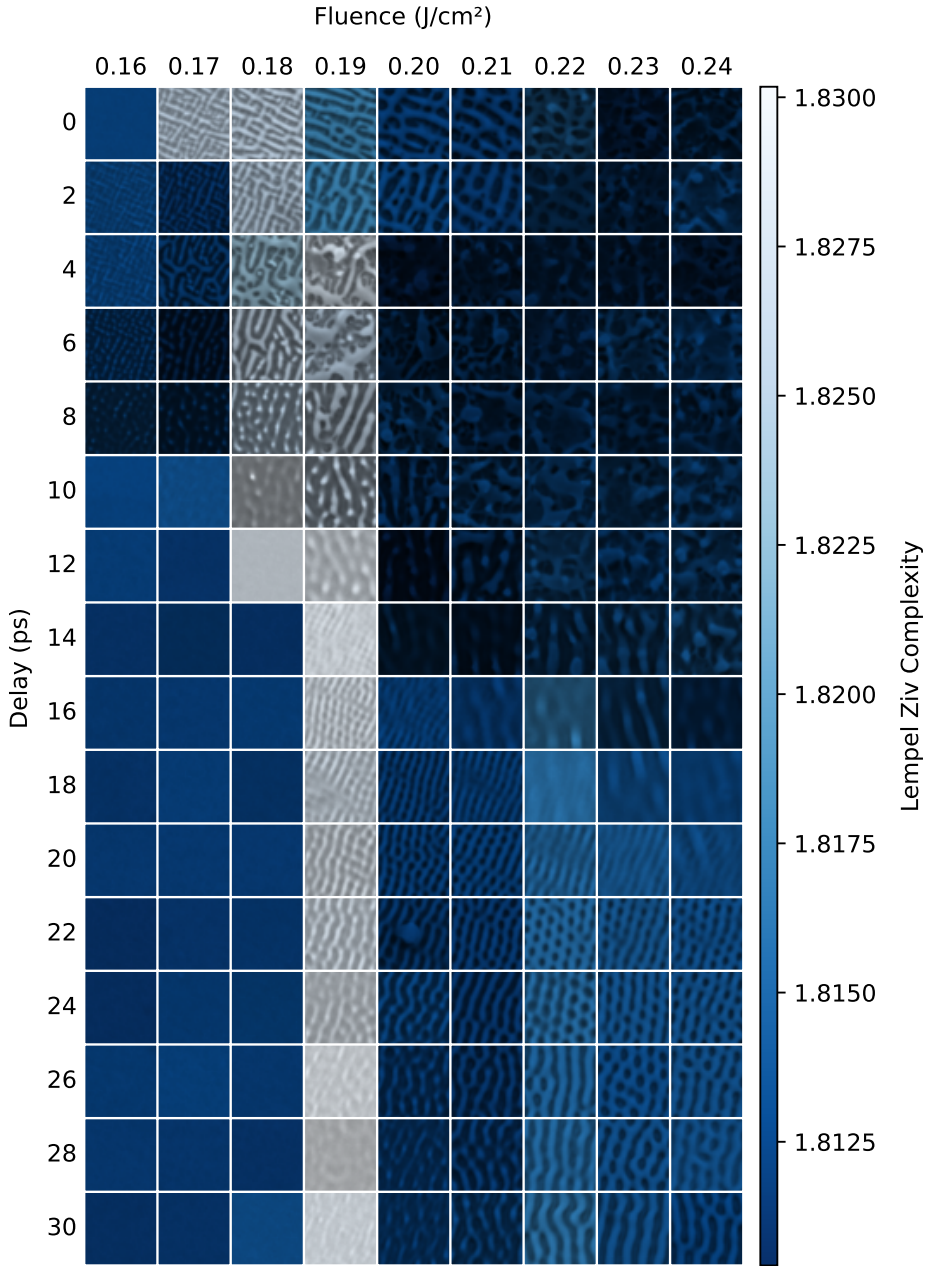


Figure A.48: Intensive Lempel-Ziv complexity of the Power spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

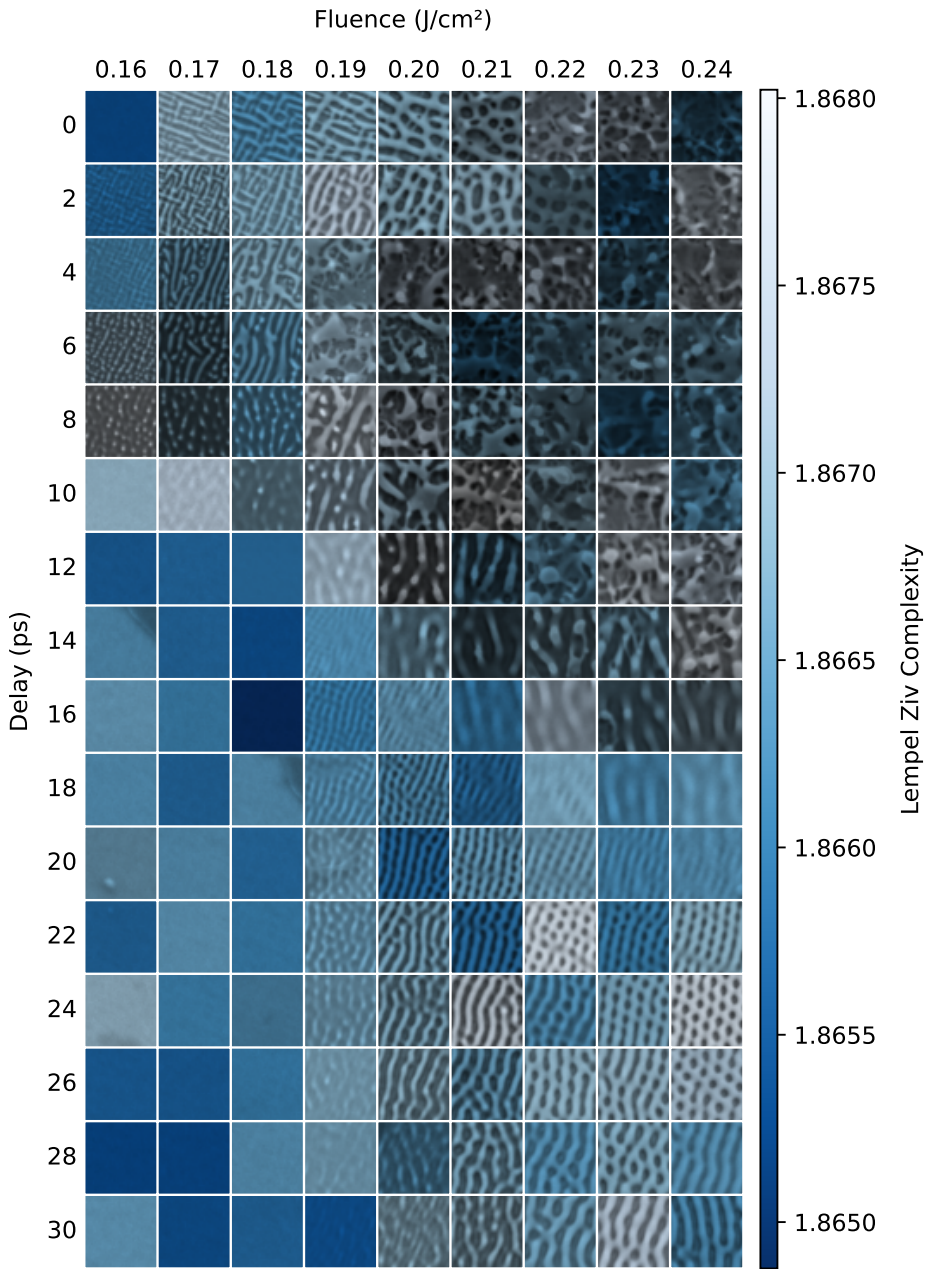


Figure A.49: Intensive Lempel-Ziv complexity of the Phase spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

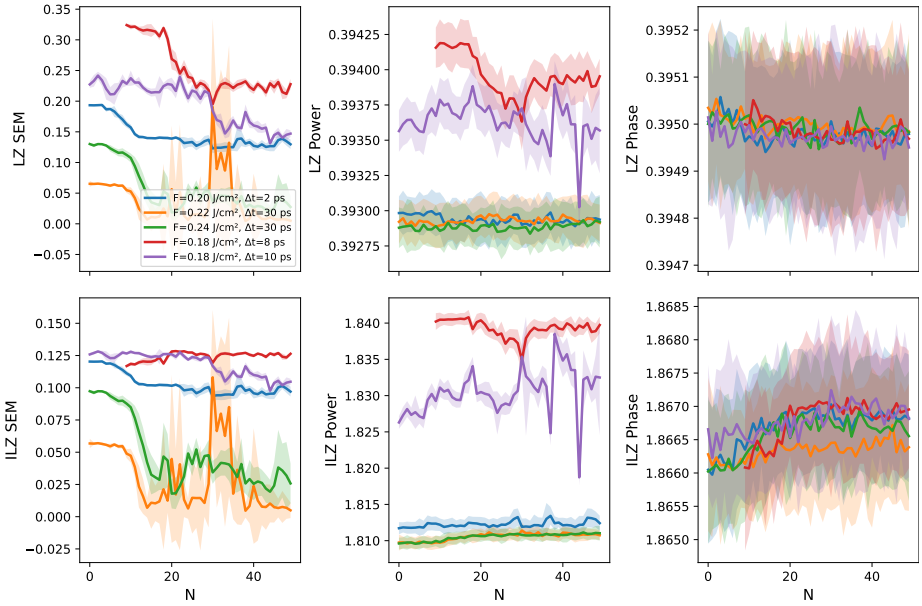


Figure A.50: Comparison Lempel-Ziv complexities on the original SEM images, their Fourier power spectra, and phase spectra(top row) and intensive complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

A.1.5 Cross-patch similarity

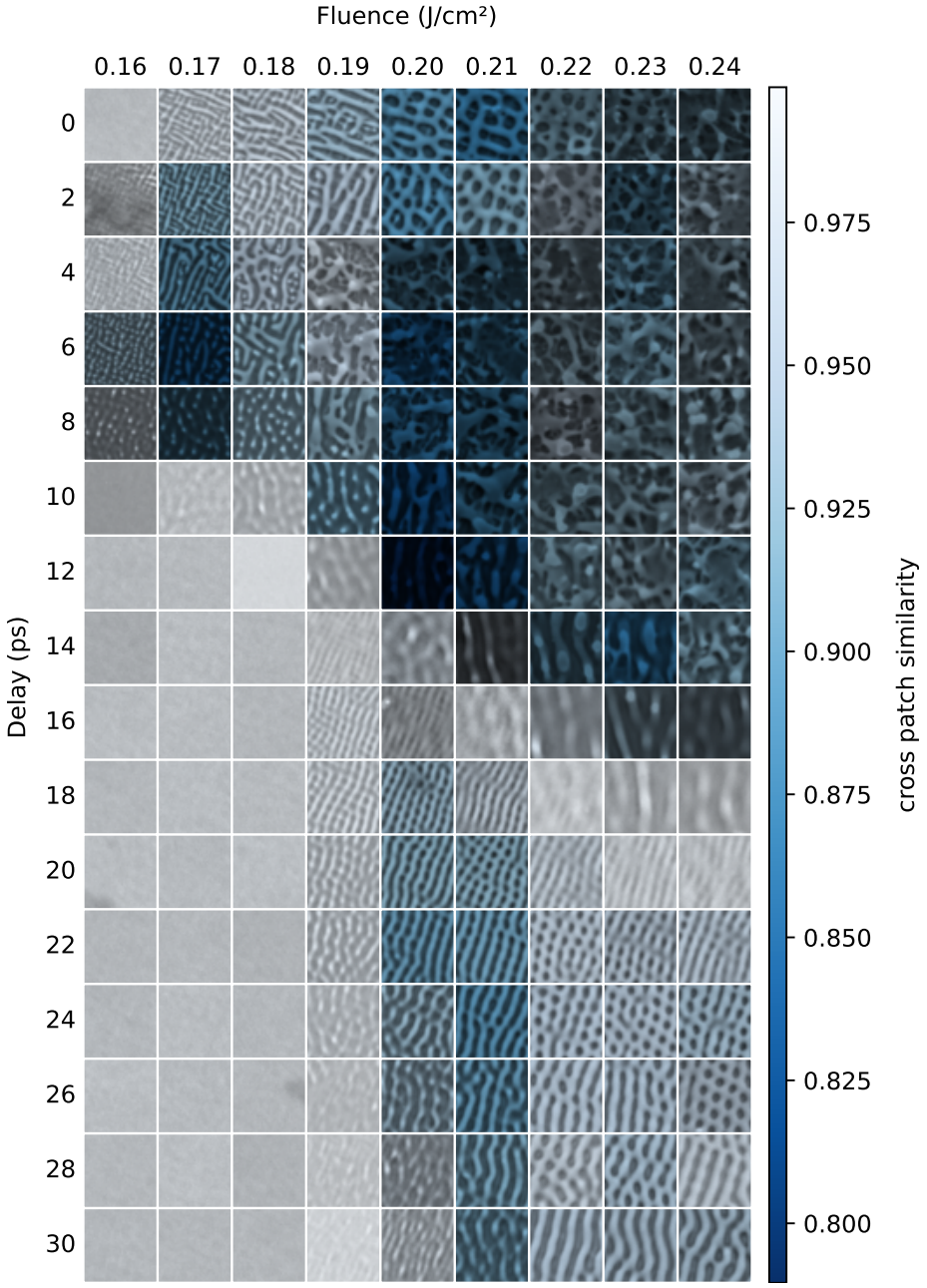


Figure A.51: Cross-patch similarity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization. Each patch is a square with side length of 56 pixels at random orientations.

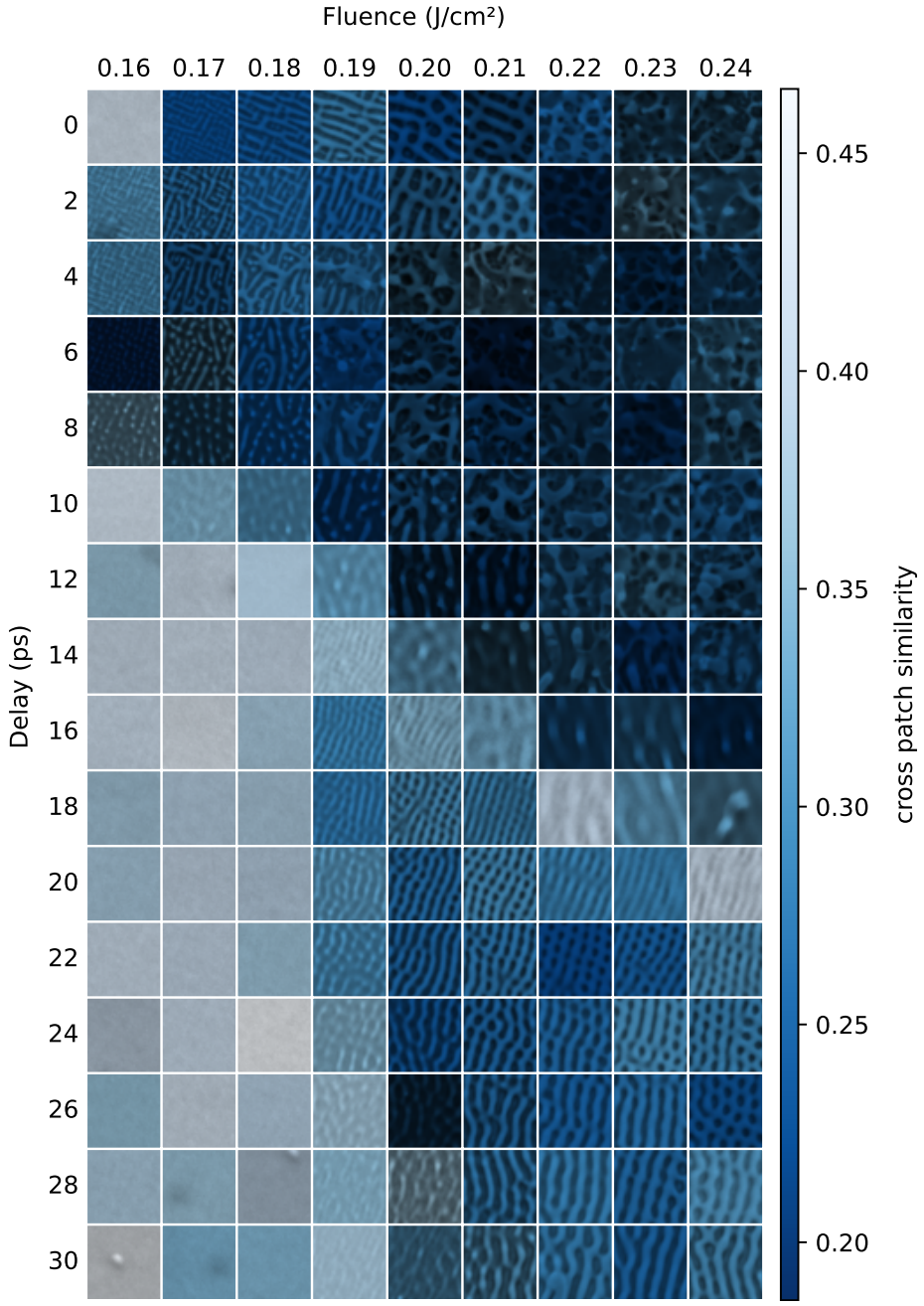


Figure A.52: Cross-patch similarity of the Power spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization. Each patch is a square with side length of 56 pixels at random orientations.

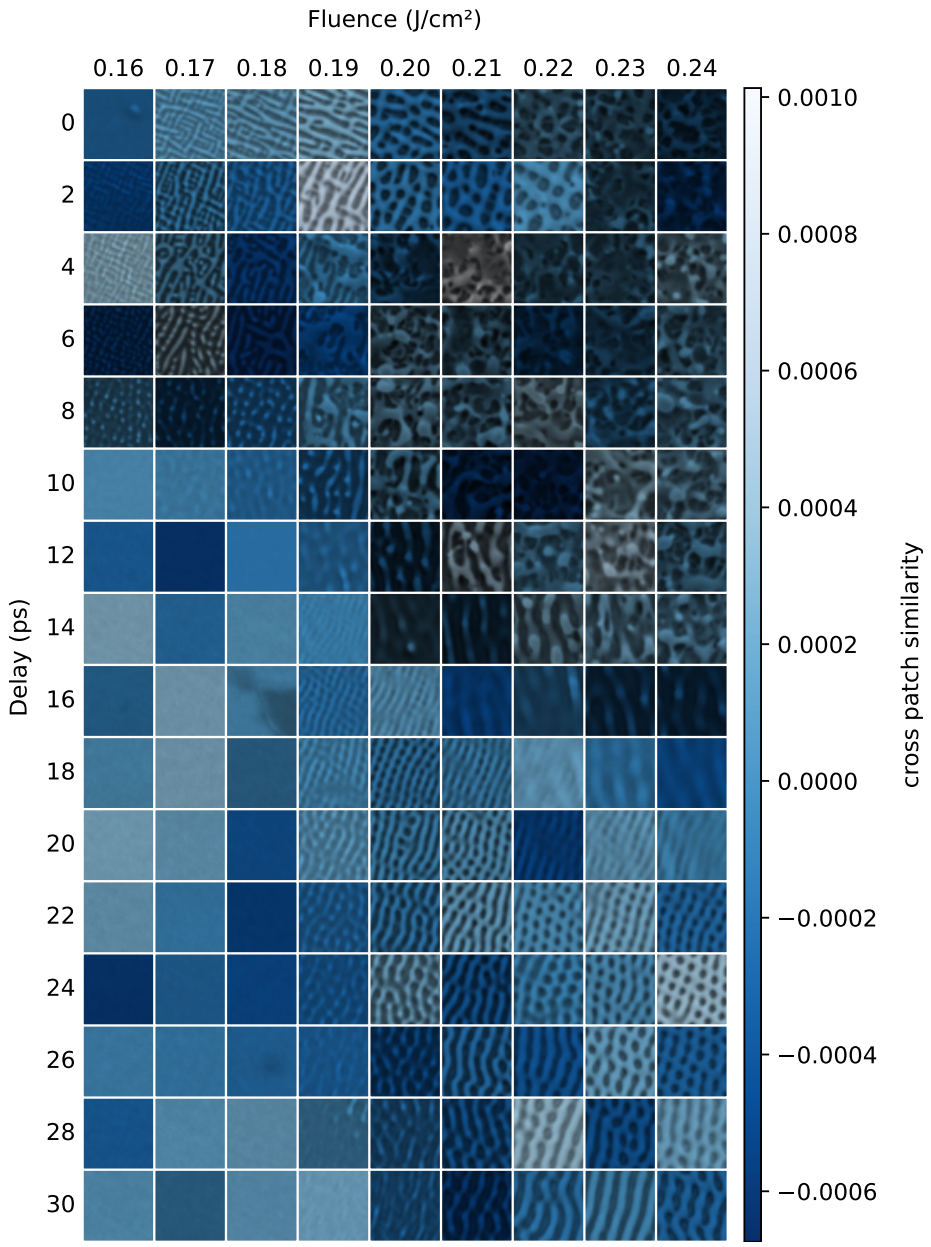


Figure A.53: Cross-patch similarity of the Phase spectrum for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization. Each patch is a square with side length of 56 pixels at random orientations.

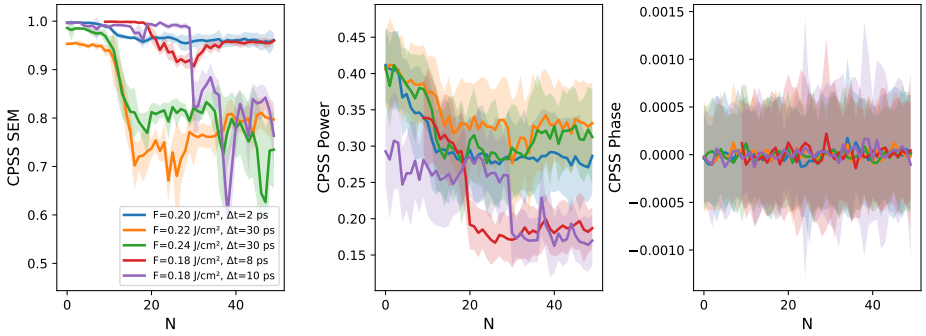


Figure A.54: Comparison of Cross-patch similarity of the original SEM images, their Fourier power spectra, and phase spectra, for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image. Each patch is a square with side length of 56 pixels at random orientations.

A.1.6 Taylor complexities

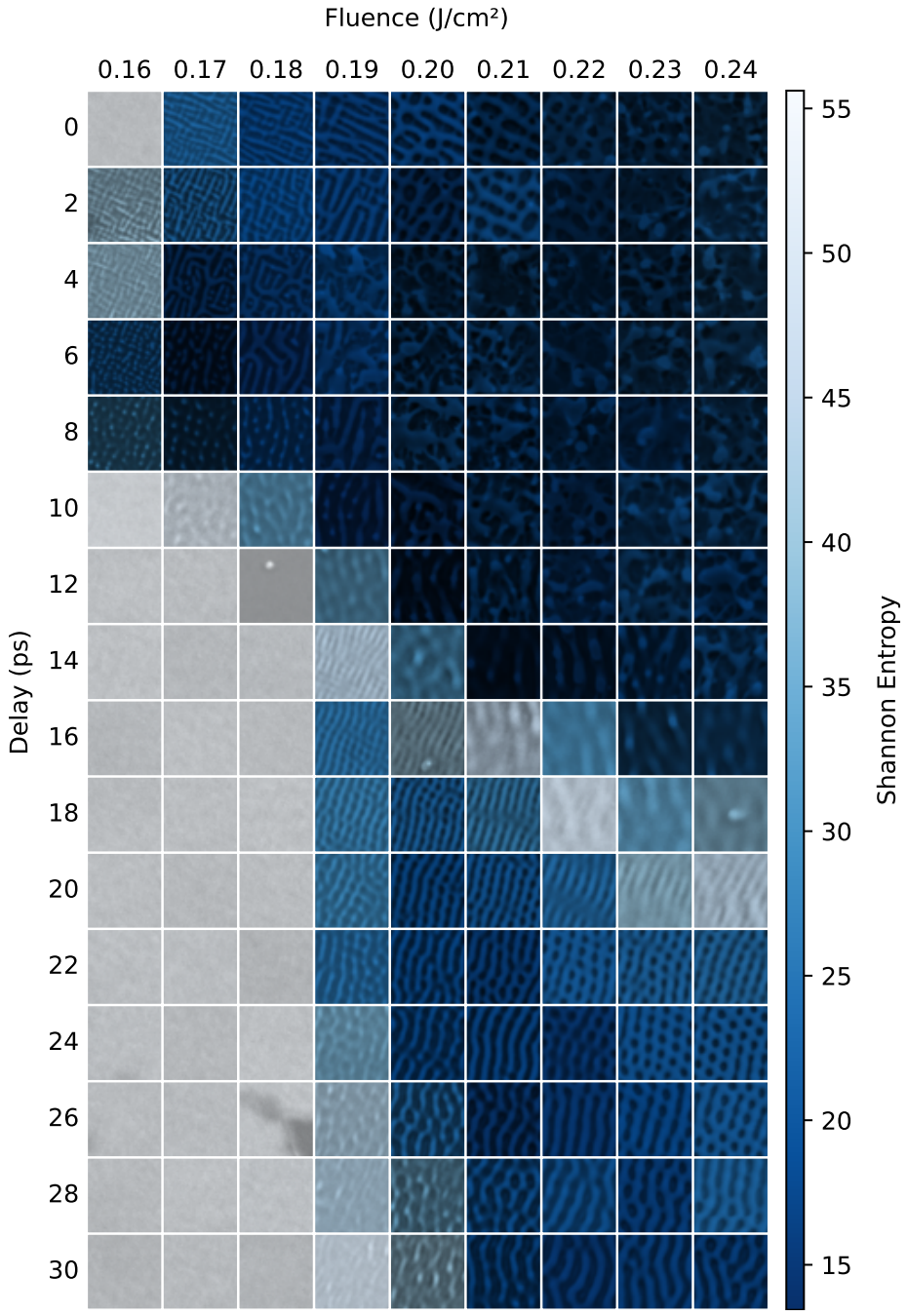


Figure A.55: Taylor Shannon entropy of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

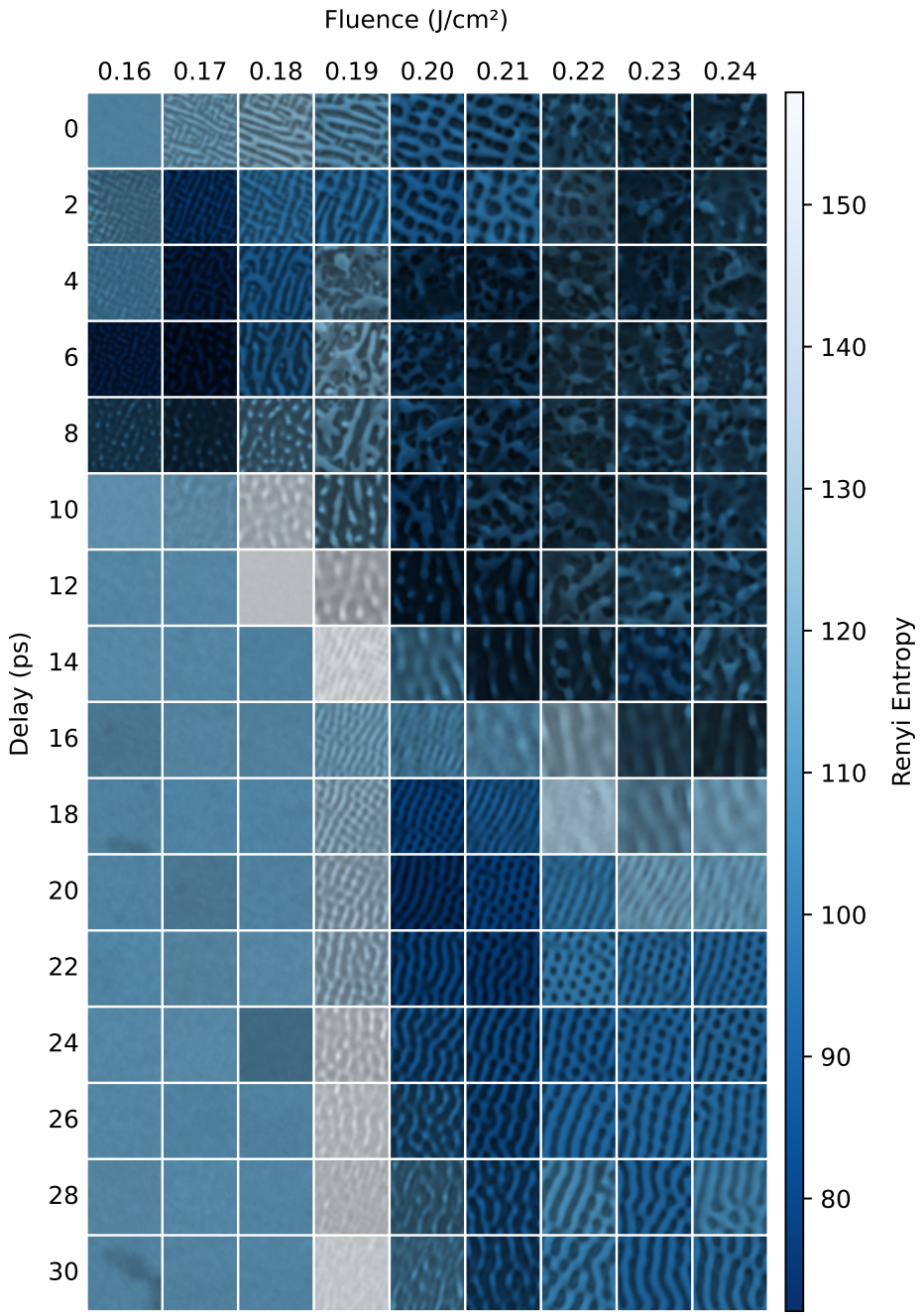


Figure A.56: Taylor Rényi entropy of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

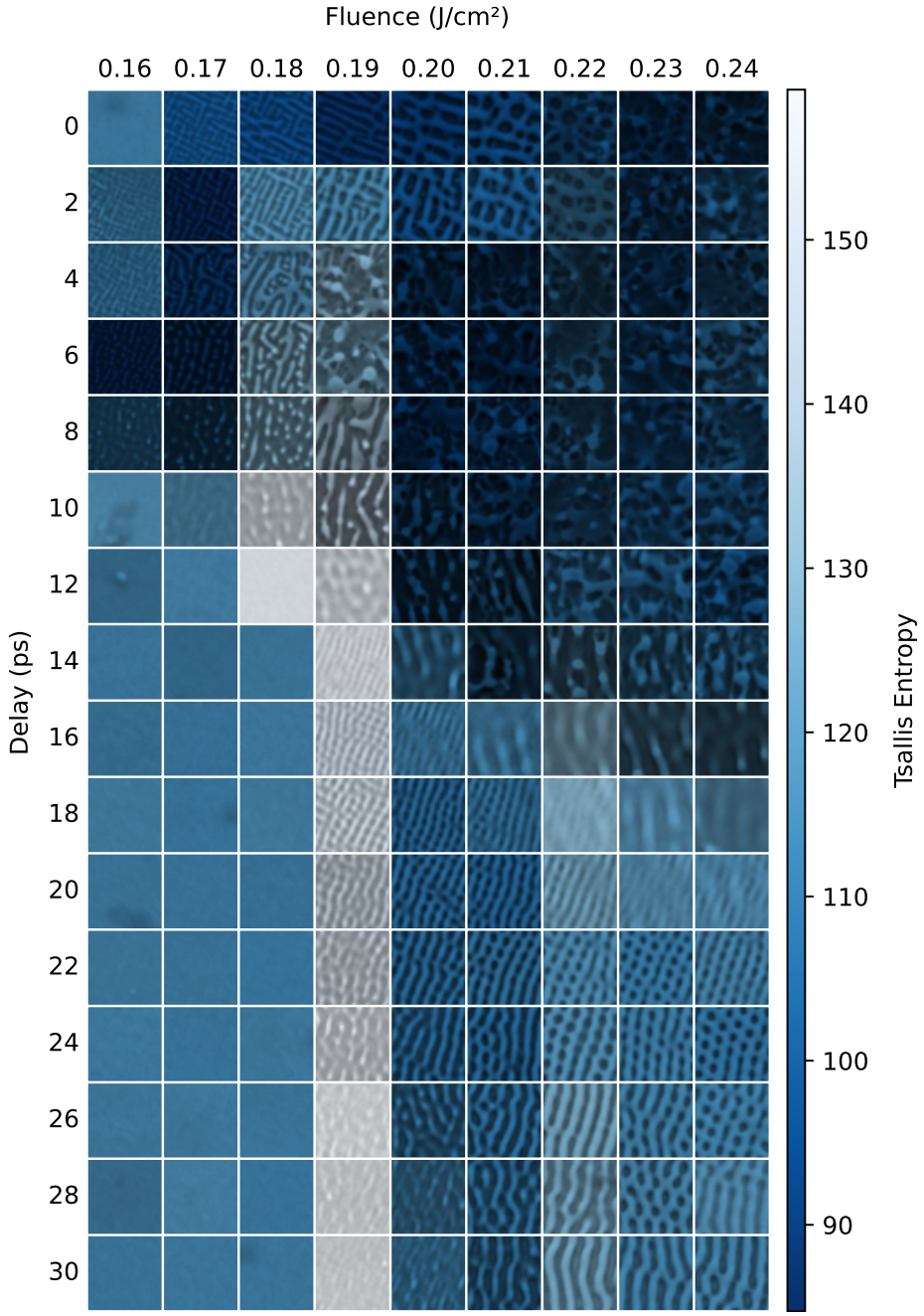


Figure A.57: Taylor Tsallis entropy of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

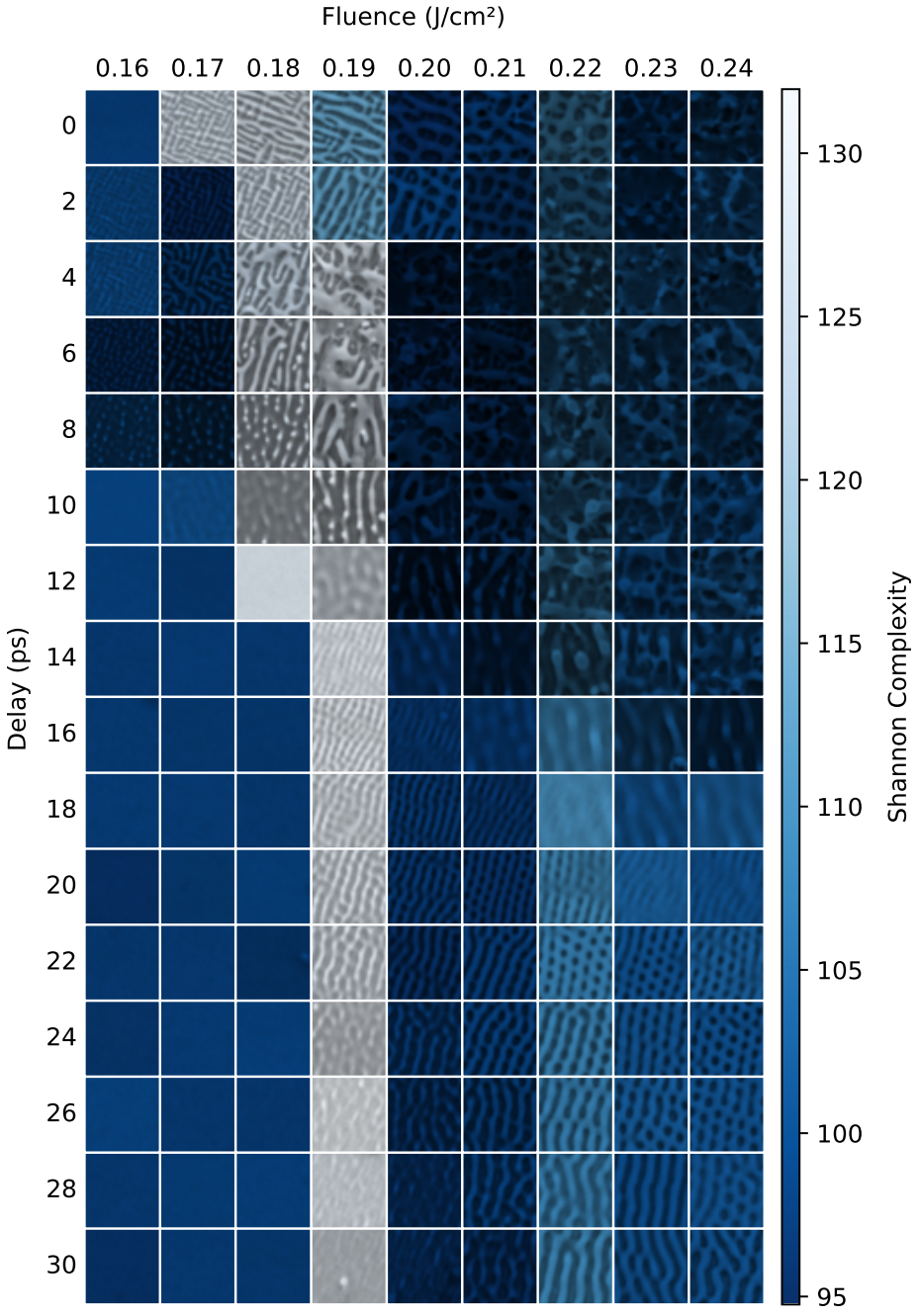


Figure A.58: Taylor Shannon complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

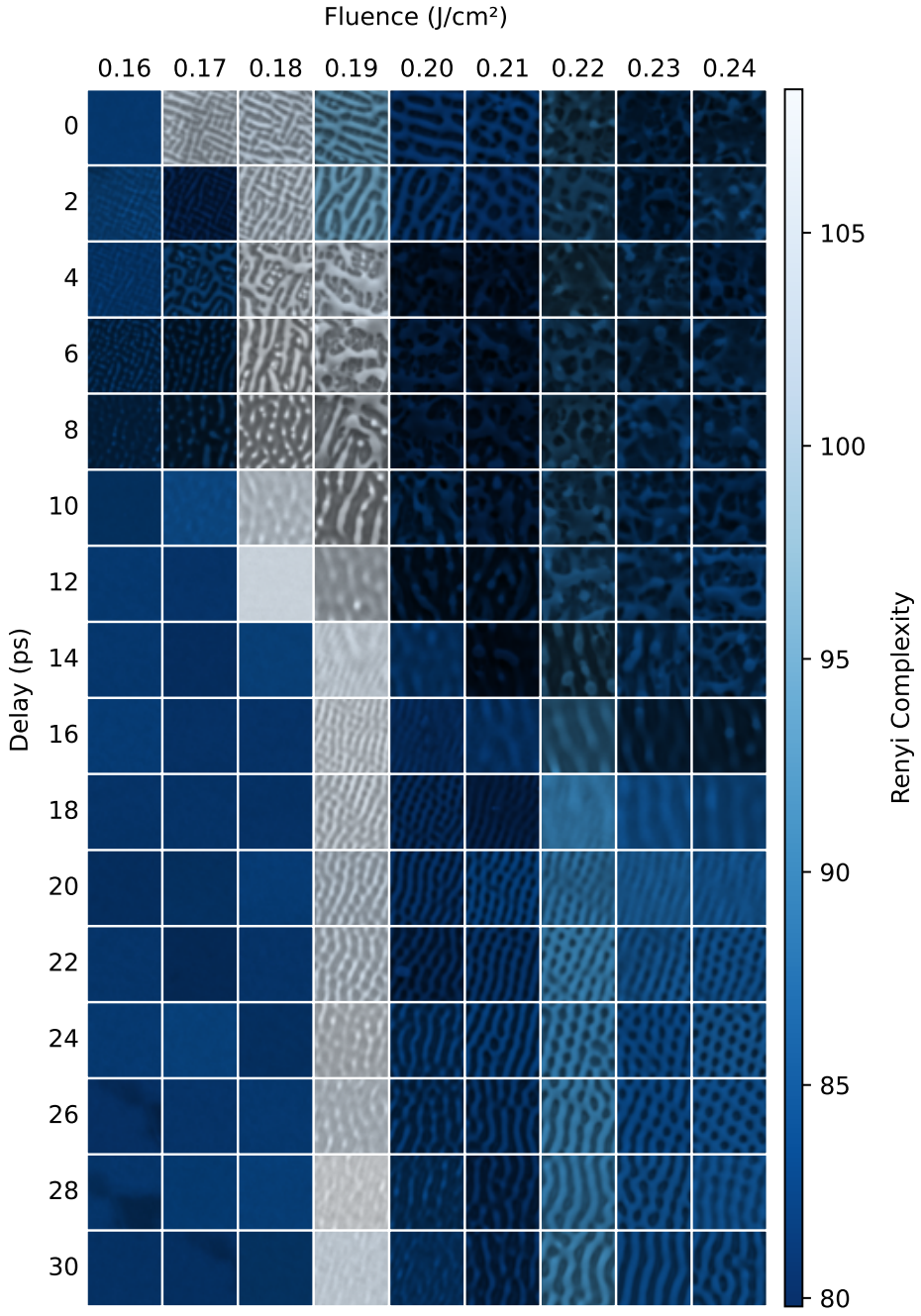


Figure A.59: Taylor Rényi Complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

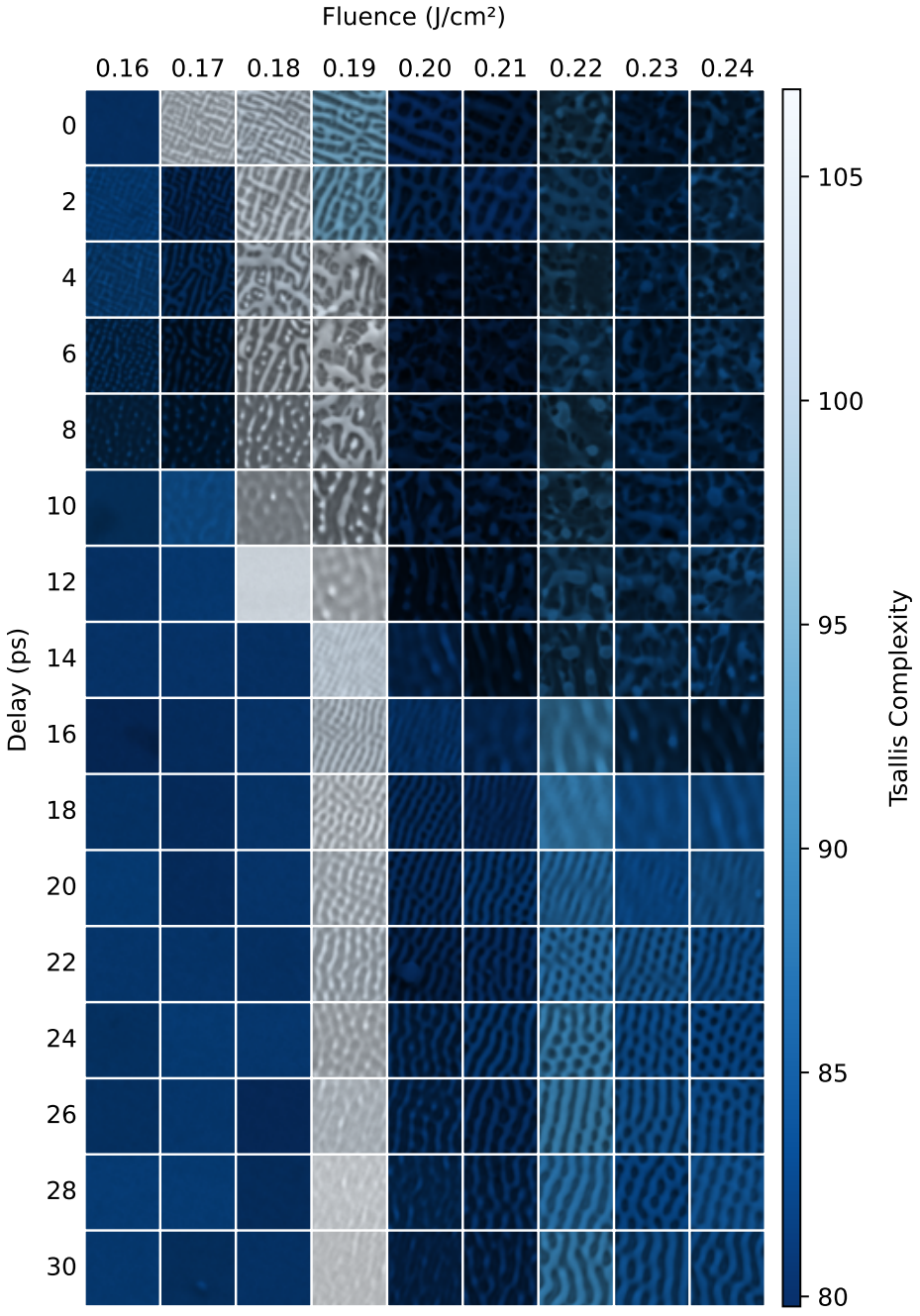


Figure A.60: Taylor Tsallis Complexity of SEM images for a range of Fluence, delay pairs and $N = 25$, as a heatmap superimposed on the original images for ease of visualization.

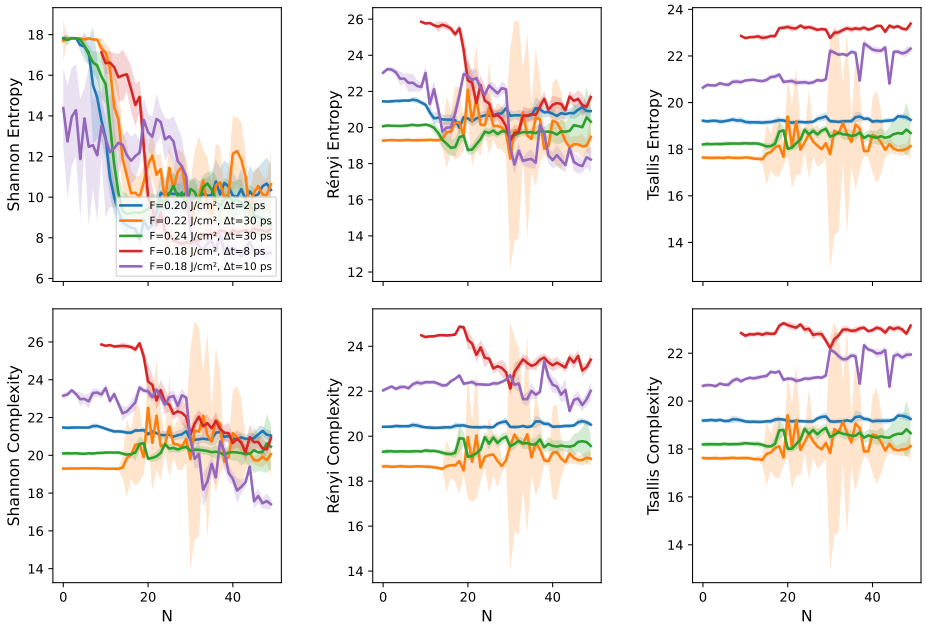


Figure A.61: Comparison of Taylor-Shannon, Rényi ($\alpha = 0.5$), and Tsallis ($q = 1.05$) entropies (top row) and complexities (bottom row) for five different experimental N series (laser parameters in the inset in the top leftmost plot, cf. Fig. A.1 for a visualization). The 2σ error bars are obtained by sampling each series of 512 square pixel images 100 times from the original SEM image.

Bibliography

[Bén00] Henri Bénard. "Les tourbillons cellulaires dans une nappe liquide". French. In: *Revue Générale des Sciences Pures et Appliquées* 11 (1900), pp. 1261–1271, 1309–1328.

[Ein05] Albert Einstein. "Zur elektrodynamik bewegter körper". In: *Annalen der physik* 4 (1905).

[Ray16] Lord Rayleigh. "On the convective currents in a horizontal layer of fluid when the higher temperature is on the under side". In: *Philosophical Magazine*. 6th series 32.192 (1916), pp. 529–546.

[Sha48] Claude E Shannon. "A Mathematical Theory of Communication". In: *Bell system technical journal* 27.3 (1948), pp. 379–423.

[Kra49] Leon Gordon Kraft. "A device for quantizing, grouping, and coding amplitude-modulated pulses". Master's thesis. Massachusetts Institute of Technology, 1949. URL: https://dspace.mit.edu/bitstream/handle/1721.1/12390/29296940-MIT.pdf.

[AK57] I Aleksandr and Akovlevich Khinchin. *Mathematical Foundations of Information Theory*. Vol. 434. Courier Corporation, 1957.

[Jay57a] Edwin T Jaynes. "Information Theory and Statistical Mechanics". In: *Physical review* 106.4 (1957), p. 620.

[Jay57b] Edwin T Jaynes. "Information Theory and Statistical Mechanics. Ii". In: *Physical review* 108.2 (1957), p. 171.

[Khi57] AI Khinchin. "Mathematical foundations of information theory dover publications inc". In: *New York* (1957).

[Pea58] J. R. A. Pearson. "On convection cells induced by surface tension". In: *J. Fluid Mech.* 4.5 (Sept. 1958), pp. 489–500. ISSN: 1469-7645. DOI: 10.1017/S0022112058000616.

[Rén+61] Alfréd Rényi et al. "On measures of entropy and information". In: *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 547-561. Berkeley, California, USA. 1961.

[PV63] Ilya Prigogine and Pierre Van Rysselberghe. "Introduction to thermodynamics of irreversible processes". In: *Journal of The Electrochemical Society* 110.4 (1963), p. 97C.

[SW63] C.E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, 1963. ISBN: 0252725484. URL: https://monoskop.org/images/b/be/Shannon_Claude_E_Weaver_Warren_The_Mathematical_Theory_of_Communication_1963.pdf.

[Sol64] RJ Solomonov. "A formal theory of inductive inference, I, II". In: *Inf. Control* 7 (1964), pp. 1–22.

[CT65] James W Cooley and John W Tukey. "An algorithm for the machine calculation of complex Fourier series". In: *Mathematics of computation* 19.90 (1965), pp. 297–301.

[Jay65] Edwin T Jaynes. "Gibbs vs Boltzmann entropies". In: *American Journal of Physics* 33.5 (1965), pp. 391–398.

[Kol65] Andrei N Kolmogorov. "Three approaches to the quantitative definition ofinformation'". In: *Problems of information transmission* 1.1 (1965), pp. 1–7.

[Str68] Gilbert Strang. "On the construction and comparison of difference schemes". In: *SIAM journal on numerical analysis* 5.3 (1968), pp. 506–517.

[Cha69] Gregory J Chaitin. "On the length of programs for computing finite binary sequences: statistical considerations". In: *Journal of the ACM (JACM)* 16.1 (1969), pp. 145–159.

[Die69] Jean Dieudonné. *Foundations of Modern Analysis, Enlarged and corrected printing.* Academic Press, 1969.

[Anu70] Paul E Anuta. "Spatial registration of multispectral and multitemporal digital imagery using fast Fourier transform techniques". In: *IEEE transactions on Geoscience Electronics* 8.4 (1970), pp. 353–368.

[Tho72] Robert C Thompson. "Principal Submatrices Ix: Interlacing Inequalities for Singular Values of Submatrices". In: *Linear Algebra and its Applications* 5.1 (1972), pp. 1–12.

[AKP+74] SI Anisimov, BL Kapeliovich, TL Perelman, et al. "Electron emission from metal surfaces exposed to ultrashort laser pulses". In: *Zh. Eksp. Teor. Fiz* 66.2 (1974), pp. 375–377.

[Gal75] Mary M Galloway. "Texture Analysis Using Gray Level Run Lengths". In: *Computer graphics and image processing* 4.2 (1975), pp. 172–179.

[KWS75] TERRENCE J Keating, PR Wolf, and FL Scarpace. "An improved method of digital image correlation". In: *Photogrammetric Engineering and Remote Sensing* 41.8 (1975), pp. 993–1002.

[LZ76] A. Lempel and J. Ziv. "On the Complexity of Finite Sequences". In: *IEEE Transactions on Information Theory* 22.1 (1976), pp. 75–81. DOI: 10.1109/TIT.1976.1055501.

[Hak77] Herman Haken. "Synergetics". In: *Physics Bulletin* 28.9 (1977), p. 412.

[Pes77] Ya B Pesin. "CHARACTERISTIC Lyapunov Exponents and Smooth Ergodic THEORY". In: *Russian Mathematical Surveys* 32.4 (Aug. 31, 1977), pp. 55–114. ISSN: 0036-0279, 1468-4829. DOI: 10.1070/RM1977v032n04ABEH001639. URL: https://doi.org/10.1070/RM1977v032n04ABEH001639 (visited on 04/11/2023).

[Siv77] GI Sivashinsky. "Nonlinear analysis of hydrodynamic instability in laminar flames—I. Derivation of basic equations". In: *Acta Astronautica* 4.11 (1977), pp. 1177–1206.

[SH77] Ju Swift and Pierre C Hohenberg. "Hydrodynamic Fluctuations At the Convective Instability". In: *Physical Review A* 15.1 (1977), p. 319.

[Jay78] E.T. Jaynes. "Where Do We Stand on Maximum Entropy?" In: *The Maximum Entropy Formalism* (1978). Ed. by R.D Levine and M. Tribus, pp. 15–118.

[Kur78] Yoshiki Kuramoto. "Diffusion-induced chaos in reaction systems". In: *Progress of Theoretical Physics Supplement* 64 (1978), pp. 346–367.

[Ris78] Jorma Rissanen. "Modeling By Shortest Data Description". In: *Automatica* 14.5 (1978), pp. 465–471.

[PM80] Y Pomeau and P Manneville. "Wavelength selection in cellular flows". In: *Physics Letters A* 75.4 (1980), pp. 296–298.

[GP83] Peter Grassberger and Itamar Procaccia. "Estimation of the Kolmogorov Entropy From a Chaotic Signal". In: *Physical review A* 28.4 (1983), p. 2591.

[Kei83] Fritz Keilmann. "Laser-driven corrugation instability of liquid metal surfaces". In: *Physical review letters* 51.23 (1983), p. 2097.

[Man83] P Manneville. "A two-dimensional model for three-dimensional convective patterns in wide containers". In: *Journal de Physique* 44.7 (1983), pp. 759–765.

[Ris83] Jorma Rissanen. "A Universal Prior for Integers and Estimation By Minimum Description Length". In: *The Annals of statistics* 11.2 (1983), pp. 416–431.

[Sip+83] JE Sipe et al. "Laser-induced periodic surface structure. I. Theory". In: *Physical Review B* 27.2 (1983), p. 1141.

[SD83] Marc K. Smith and Stephen H. Davis. "Instabilities of dynamic thermocapillary liquid layers. Part 1. Convective instabilities". In: *J. Fluid Mech.* 132 (July 1983), pp. 119–144. ISSN: 1469-7645. DOI: 10.1017/S0022112083001512.

[YSV84] Jeff F Young, JE Sipe, and HM Van Driel. "Laser-induced periodic surface structure. III. Fluence regimes, the role of feedback, and details of the induced topography in germanium". In: *Physical Review B* 30.4 (1984), p. 2001.

[ER85] J-P Eckmann and David Ruelle. "Ergodic Theory of Chaos and Strange Attractors". In: *Reviews of modern physics* 57.3 (1985), p. 617.

[BQG86] Gerd Binnig, Calvin F Quate, and Ch Gerber. "Atomic force microscope". In: *Physical review letters* 56.9 (1986), p. 930.

[Ris86] Jorma Rissanen. "Stochastic complexity and modeling". In: *The annals of statistics* (1986), pp. 1080–1100.

[Smi86] Marc K. Smith. "Instability mechanisms in dynamic thermocapillary liquid layers". In: *Phys. Fluids* 29.10 (Oct. 1986), pp. 3182–3186. ISSN: 0031-9171. DOI: 10.1063/1.865836.

[Ris87] Jorma Rissanen. "Stochastic complexity". In: *Journal of the Royal Statistical Society: Series B (Methodological)* 49.3 (1987), pp. 223–239.

[Sht87] Yurii Mikhailovich Shtar'kov. "Universal sequential coding of single messages". In: *Problemy Peredachi Informatsii* 23.3 (1987), pp. 3–17.

[Ben88] Charles H Bennett. "Logical depth and physical complexity". In: *A half-century survey on The Universal Turing Machine*. 1988, pp. 227–257.

[Cor+88] PB Corkum et al. "Thermal response of metals to ultrashort-pulse laser excitation". In: *Physical review letters* 61.25 (1988), p. 2886.

[Tsa88] Constantino Tsallis. "Possible Generalization of Boltzmann-Gibbs Statistics". In: *Journal of statistical physics* 52 (1988), pp. 479–487.

[Cyb89] George Cybenko. "Approximation by superpositions of a sigmoidal function". In: *Mathematics of Control, Signals, and Systems* 2.4 (1989), pp. 303–314.

[HSW89] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators". In: *Neural Networks* 2.5 (1989), pp. 359–366.

[Rue89] David Ruelle. *Chaotic evolution and strange attractors*. Vol. 1. Cambridge University Press, 1989.

[WZ89] Aaron D Wyner and Jacob Ziv. "Some asymptotic properties of the entropy of a stationary ergodic data source with applications to data compression". In: *IEEE Transactions on Information Theory* 35.6 (1989), pp. 1250–1258.

[Yos90] Haruo Yoshida. "Construction of Higher Order Symplectic Integrators". In: *Physics letters A* 150.5-7 (1990), pp. 262–268.

[ABK91] Henry DI Abarbanel, Reggie Brown, and Matthew B Kennel. "Variation of Lyapunov Exponents on a Strange Attractor". In: *Journal of Nonlinear Science* 1 (1991), pp. 175–199.

[Bar91] Andrew R Barron. "Complexity regularization with application to artificial neural networks". In: *Nonparametric functional estimation and related topics* (1991), pp. 561–576.

[ABK92] Henry DI Abarbanel, Reggie Brown, and Matthew B Kennel. "Local Lyapunov Exponents Computed From Observed Data". In: *Journal of Nonlinear Science* 2 (1992), pp. 343–365.

[EVG92] KR Elder, Jorge Vinals, and Martin Grant. "Ordering dynamics in the two-dimensional stochastic Swift-Hohenberg equation". In: *Physical review letters* 68.20 (1992), p. 3024.

[Jay92] E. T. Jaynes. "The Gibbs Paradox". In: *Maximum Entropy and Bayesian Methods: Seattle, 1991*. Ed. by C. Ray Smith, Gary J. Erickson, and Paul O. Neudorfer. Dordrecht: Springer Netherlands, 1992, pp. 1–21. ISBN: 978-94-017-2219-3. DOI: 10.1007/978-94-017-2219-3_1. URL: https://doi.org/10.1007/978-94-017-2219-3_1.

[KK92] Jagat Narain Kapur and Hiremaglur K Kesavan. "Entropy optimization principles and their applications". In: *Entropy and energy dissipation in water resources*. Springer, 1992, pp. 3–20.

[KH92] Anders Krogh and John A Hertz. "A simple weight decay can improve generalization". In: *Advances in neural information processing systems*. 1992, pp. 950–957.

[PM92] William B Pennebaker and Joan L Mitchell. *JPEG: Still image data compression standard*. Springer Science & Business Media, 1992.

[CH93] Mark C Cross and Pierre C Hohenberg. "Pattern Formation Outside of Equilibrium". In: *Reviews of modern physics* 65.3 (1993), p. 851.

[HV93] Geoffrey E Hinton and Drew Van Camp. "Keeping the neural networks simple by minimizing the description length of the weights". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 5–13.

[OW93] Donald S Ornstein and Benjamin Weiss. "Entropy and data compression schemes". In: *IEEE Transactions on information theory* 39.1 (1993), pp. 78–83.

[SW93] N.J.A. Sloane and A.D. Wyner, eds. *Claude Elwood Shannon: Collected Papers*. New York, NY: IEEE Press, 1993.

[DPW94] Werner Decker, Werner Pesch, and Andreas Weber. "Spiral defect chaos in Rayleigh-Bénard convection". In: *Physical review letters* 73.5 (1994), p. 648.

[LBF94] Igor'V. Lomonosov, Aleksey V. Bushman, and Vladimir E. Fortov. "Equations of state for metals at high energy densities". In: *AIP Conference Proceedings*. Vol. 309. AIP. 1994, pp. 117–120.

[VHV94] M Van Hecke, PC Hohenberg, and W Van Saarloos. "Amplitude equations for pattern forming systems". In: *Fundamental problems in statistical mechanics VIII* (1994), pp. 245–278.

[BS95] Christian Beck and Friedrich Schögl. *Thermodynamics of chaotic systems*. 1995.

[LMC95] Ricardo Lopez-Ruiz, Hector L Mancini, and Xavier Calbet. "A Statistical Measure of Complexity". In: *Physics letters A* 209.5-6 (1995), pp. 321–326.

[TB95] A. Thess and M. Bestehorn. "Planform selection in Bénard-Marangoni convection: l hexagons versus g hexagons". In: *Phys. Rev. E* 52.6 (Dec. 1995), pp. 6358–6367. ISSN: 2470-0053. DOI: 10.1103/PhysRevE.52.6358.

[Kob+96] Daisuke Kobayashi et al. "Extracting speech features from human speech like noise". In: *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*. Vol. 1. IEEE. 1996, pp. 418–421.

[San96] Marco Sandri. "Numerical Calculation of Lyapunov Exponents". In: *Mathematica Journal* 6.3 (1996), pp. 78–84.

[Edm97] Bruce Edmonds. "Hypertext bibliography of measures of complexity". In: *URL http://www. cpm. mmu. ac. uk/ bruce/combib* (1997).

[Ris97] J. Rissanen. "Stochastic Complexity in Learning". In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 89–95. ISSN: 0022-0000. DOI: https://doi.org/10.1006/jcss.1997.1501. URL: https://www.sciencedirect.com/science/article/pii/S0022000097915014.

[BRY98] Andrew Barron, Jorma Rissanen, and Bin Yu. "The Minimum Description Length Principle in Coding and Modeling". In: *IEEE transactions on information theory* 44.6 (1998), pp. 2743–2760.

[BV98] J. Bragard and M. G. Velarde. "Bénard-Marangoni convection: planforms and related theoretical predictions". In: *J. Fluid Mech.* 368 (Aug. 1998), pp. 165–194. ISSN: 1469-7645. DOI: 10.1017/S0022112098001669.

[Her+98] Tsing-Hua Her et al. "Microstructuring of silicon with femtosecond laser pulses". In: *Applied Physics Letters* 73.12 (1998), pp. 1673–1675.

[Kon+98] Ioannis Kontoyiannis et al. "Nonparametric entropy estimation for stationary processes and random fields, with applications to English text". In: *IEEE Transactions on Information Theory* 44.3 (1998), pp. 1319–1327.

[LLF98] Isaac E Lagaris, Aristidis Likas, and Dimitrios I Fotiadis. "Artificial neural networks for solving ordinary and partial differential equations". In: *IEEE transactions on neural networks* 9.5 (1998), pp. 987–1000.

[Yam98] Kenji Yamanishi. "A decision-theoretic extension of stochastic complexity and its applications to learning". In: *IEEE Transactions on Information Theory* 44.4 (1998), pp. 1424–1439.

[BT99] Thomas Boeck and Andr Thess. "Bnard–Marangoni convection at low Prandtl number". In: *J. Fluid Mech.* 399 (Nov. 1999), pp. 251–275. ISSN: 1469-7645. DOI: 10.1017/S0022112099006436.

[Abe00] Sumiyoshi Abe. "Axioms and Uniqueness Theorem for Tsallis Entropy". In: *Physics Letters A* 271.1-2 (2000), pp. 74–79.

[BPA00] Eberhard Bodenschatz, Werner Pesch, and Guenter Ahlers. "Recent Developments in Rayleigh-Bnard Convection". In: *Annu. Rev. Fluid Mech.* 32.1 (Jan. 2000), pp. 709–778. ISSN: 0066-4189. DOI: 10.1146/annurev.fluid.32.1.709.

[ER00] Blas Echebarria and Hermann Riecke. "Defect chaos of oscillating hexagons in rotating convection". In: *Physical review letters* 84.21 (2000), p. 4838.

[Ego+00] David A Egolf et al. "Mechanisms of extensive spatiotemporal chaos in Rayleigh–Bénard convection". In: *Nature* 404.6779 (2000), pp. 733–736.

[Rei00] Ludwig Reimer. "Scanning electron microscopy: physics of image formation and microanalysis". In: *Measurement Science and Technology* 11.12 (2000), pp. 1826–1826.

[Ris00] Jorma Rissanen. "Mdl Denoising". In: *IEEE Transactions on Information Theory* 46.7 (2000), pp. 2537–2543.

[VL00] Paul MB Vitányi and Ming Li. "Minimum Description Length Induction, Bayesianism, and Kolmogorov Complexity". In: *IEEE Transactions on information theory* 46.2 (2000), pp. 446–464.

[FHT+01] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York, 2001.

[KO01] Marc C Kennedy and Anthony O'Hagan. "Bayesian Calibration of Computer Models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63.3 (2001), pp. 425–464.

[Llo01] Seth Lloyd. "Measures of complexity: a nonexhaustive list". In: *IEEE Control Systems Magazine* 21.4 (2001), pp. 7–8.

[Ris01] Jorma Rissanen. "Strong Optimality of the Normalized Ml Models As Universal Codes and Information in Data". In: *IEEE Transactions on Information Theory* 47.5 (2001), pp. 1712–1717.

[Sha01] Cosma Rohilla Shalizi. "Causal architecture, complexity and self-organization in time series and cellular automata". PhD thesis. The University of Wisconsin-Madison, 2001.

[BKP02] Christoph Bandt, Gerhard Keller, and Bernd Pompe. "Entropy of Interval Maps Via Permutations". In: *Nonlinearity* 15.5 (Sept. 1, 2002), pp. 1595–1602. ISSN: 09517715. DOI: 10.1088/0951-7715/15/5/312. URL: https://doi.org/10.1088/0951-7715/15/5/312 (visited on 04/11/2023).

[BP02] Christoph Bandt and Bernd Pompe. "Permutation Entropy: a Natural Complexity Measure for Time Series". In: *Physical review letters* 88.17 (2002), p. 174102.

[BL03] Avrim Blum and John Langford. "Pac-mdl bounds". In: *Learning Theory and Kernel Machines: 16th Annual Conference on Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24-27, 2003. Proceedings*. Springer. 2003, pp. 344–357.

[Hal03] Brian C Hall. "The Baker-Campbell-Hausdorff Formula". In: *Lie Groups, Lie Algebras, and Representations*. Springer, 2003, pp. 63–90.

[HY03] Mark H Hansen and Bin Yu. "Minimum Description Length Model Selection Criteria for Generalized Linear Models". In: *Lecture Notes-Monograph Series* (2003), pp. 145–163.

[MM03] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[Shi+03] Yasuhiko Shimotsuma et al. "Self-organized nanogratings in glass irradiated by ultrashort light pulses". In: *Physical review letters* 91.24 (2003), p. 247405.

[GL04] Murray Gell-Mann and Seth Lloyd. "Effective Complexity". In: *Nonextensive Entropy: Interdisciplinary Applications*. Oxford University Press, Apr. 2004. ISBN: 9780195159769. DOI: 10.1093/oso/9780195159769.003.0028. eprint: https://academic.oup.com/book/0/chapter/354534715/chapter-pdf/43717941/isbn-9780195159769-book-part-28.pdf. URL: https://doi.org/10.1093/oso/9780195159769.003.0028.

[GT04] Murray Gell-Mann and Constantino Tsallis. *Nonextensive entropy: interdisciplinary applications*. Oxford University Press, 2004.

[GC04] Stefan Gheorghiu and Marc-Olivier Coppens. "Heterogeneity Explains Features of "Anomalous" Thermodynamics and Statistics". In: *Proceedings of the National Academy of Sciences* 101.45 (2004), pp. 15852–15856.

[Li+04] Ming Li et al. "The similarity metric". In: *IEEE transactions on Information Theory* 50.12 (2004), pp. 3250–3264.

[AKK05] Jose M. Amigo, Matthew B. Kennel, and Ljupco Kocarev. "The Permutation Entropy Rate Equals the Metric Entropy Rate for Ergodic Information Sources and Ergodic Dynamical Systems". In: *Physica D: Nonlinear Phenomena* 210.1-2 (Oct. 2005), pp. 77–95. ISSN: 01672789. DOI: 10.1016/j.physd.2005.07.006. arXiv: nlin/0503044. URL: https://doi.org/10.1016/j.physd.2005.07.006 (visited on 04/11/2023).

[Cay05] Lawrence Cayton. "Algorithms for manifold learning". In: *Univ. of California at San Diego Tech. Rep* 12.1-17 (2005), p. 1.

[Grü05] Peter Grünwald. "Minimum Description Length Tutorial". In: *Advances in minimum description length: Theory and applications* 5 (2005), pp. 1–80.

[Tar05] Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, 2005.

[ZRB05] S. Zozor, P. Ravier, and O. Buttelli. "On Lempel–Ziv complexity for multidimensional data analysis". In: *Physica A: Statistical Mechanics and its Applications* 345.1 (2005), pp. 285–302. ISSN: 0378-4371. DOI: https://doi.org/10.1016/j.physa.2004.07.025. URL: https://www.sciencedirect.com/science/article/pii/S037843710400994X.

[MNP06] Jay I Myung, Daniel J Navarro, and Mark A Pitt. "Model Selection By Normalized Maximum Likelihood". In: *Journal of Mathematical Psychology* 50.2 (2006), pp. 167–179.

[Sha06] Cosma Rohilla Shalizi. "Methods and Techniques of Complex Systems Science: An Overview". In: *Complex Systems Science in Biomedicine*. Ed. by Thomas S. Deisboeck and J. Yasha Kresh. Boston, MA: Springer US, 2006, pp. 33–114. ISBN: 978-0-387-33532-2. DOI: 10.1007/978-0-387-33532-2_2. URL: https://doi.org/10.1007/978-0-387-33532-2_2.

[Var+06] Olga Varlamova et al. "Self-organized pattern formation upon femtosecond laser ablation by circularly polarized light". In: *Applied Surface Science* 252.13 (2006), pp. 4702–4706.

[GL07] Peter Grünwald and John Langford. "Suboptimal behavior of Bayes and MDL in classification under misspecification". In: *Machine Learning* 66 (2007), pp. 119–149.

[Lev07] Randall Leveque. *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems (Classics in Applied Mathematics)*. Classics in Applied Mathematics. SIAM, Society for Industrial and Applied Mathematics, 2007. ISBN: 0898716292,9780898716290. URL: http://gen.lib.rus.ec/book/index.php?md5=3fc227edd62748555b6e54b2f401d4d8.

[Ros+07] Osvaldo A Rosso et al. "Distinguishing Noise From Chaos". In: *Physical review letters* 99.15 (2007), p. 154102.

[LZC08] Zhibin Lin, Leonid V Zhigilei, and Vittorio Celli. "Electron-phonon coupling and electron heat capacity of metals under conditions of strong electron-phonon nonequilibrium". In: *Physical Review B* 77.7 (2008), p. 075133.

[Den+09] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[KH+09] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning Multiple Layers of Features From Tiny Images". In: (2009).

[SOS09] Michael A Sutton, Jean Jose Orteu, and Hubert Schreier. *Image correlation for shape, motion and deformation measurements: basic concepts, theory and applications*. Springer Science & Business Media, 2009.

[BS10] R Mark Bradley and Patrick D Shipman. "Spontaneous pattern formation induced by ion bombardment of binary compounds". In: *Physical Review Letters* 105.14 (2010), p. 145501.

[Dus+10] Benjamin Dusser et al. "Controlled Nanostructrures Formation By Ultra Fast Laser Pulses for Color Marking". In: *Optics express* 18.3 (2010), pp. 2913–2924.

[LCB10] Yann LeCun, Corinna Cortes, and CJ Burges. "Mnist Handwritten Digit Database". In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010).

[Ran+10] A Ranella et al. "Tuning cell adhesion by controlling the roughness and wettability of 3D micro/nano silicon structures". In: *Acta biomaterialia* 6.7 (2010), pp. 2711–2720.

[Sze10] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[BFL11] John C Baez, Tobias Fritz, and Tom Leinster. "A characterization of entropy in terms of information loss". In: *Entropy* 13.11 (2011), pp. 1945–1957.

[Fad+11] Elena Fadeeva et al. "Bacterial Retention on Superhydrophobic Titanium Surfaces Fabricated By Femtosecond Laser Ablation". In: *Langmuir* 27.6 (2011), pp. 3012–3019.

[Lu11] Jessica R. Lu. *Azimuthally averaged radial profile*. Sept. 2011. URL: https://www.astrobetter.com/wiki/python_radial_profiles.

[Sid11] Hira Affan Siddiqui. "Spatio-temporal Patterns in Systems Far from Equilibrium". PhD thesis. 2011.

[Tei+11] Andreia Teixeira et al. "Entropy Measures vs. Kolmogorov Complexity". In: *Entropy* 13.3 (2011), pp. 595–611. ISSN: 1099-4300. DOI: 10.3390/e13030595. URL: https://www.mdpi.com/1099-4300/13/3/595.

[ATT12] J. Argaez, D. Torres, and H. Toral. "The role of entropic index for detection of anomalous behaviors in a VoIP system". In: *CONIELECOMP 2012, 22nd International Conference on Electrical Communications and Computers*. 2012, pp. 92–96. DOI: 10.1109/CONIELECOMP.2012.6189888. URL: https://doi.org/10.1109/CONIELECOMP.2012.6189888.

[Bon+12] J. Bonse et al. "Femtosecond laser-induced periodic surface structures". In: *J. Laser Appl.* 24.4 (July 2012), p. 042006. ISSN: 1042-346X. DOI: 10.2351/1.4712658.

[BM12] Joan Bruna and Stéphane Mallat. *Invariant Scattering Convolution Networks*. 2012. DOI: 10.48550/ARXIV.1203.1513. URL: https://arxiv.org/abs/1203.1513.

[CT12] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[Den12] Li Deng. "The Mnist Database of Handwritten Digit Images for Machine Learning Research". In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 141–142.

[Gra12] Peter Grassberger. "Randomness, information, and complexity". In: *arXiv preprint arXiv:1208.3459* (2012).

[Rei+12] Juergen Reif et al. "The role of asymmetric excitation in self-organized nanostructure formation upon femtosecond laser ablation". In: *AIP Conference Proceedings*. Vol. 1464. 1. American Institute of Physics. 2012, pp. 428–441.

[Rib+12] Haroldo V Ribeiro et al. "Complexity-Entropy Causality Plane As a Complexity Measure for Two-Dimensional Patterns". In: (2012).

[Wal12] Daniel Walgraef. *Spatio-temporal pattern formation: with examples from physics, chemistry, and materials science*. Springer Science & Business Media, 2012.

[Yao+12] Jianwu Yao et al. "Selective Appearance of Several Laser-Induced Periodic Surface Structure Patterns on a Metal Surface Using Structural Colors Produced By Femtosecond Laser Pulses". In: *Applied Surface Science* 258.19 (2012), pp. 7625–7632.

[CV13] Marcel G Clerc and Nicolas Verschueren. "Quasiperiodicity route to spatiotemporal chaos in one-dimensional pattern-forming systems". In: *Physical Review E* 88.5 (2013), p. 052916.

[Fad+13] Bilal Fadlallah et al. "Weighted-Permutation Entropy: A Complexity Measure for Time Series Incorporating Amplitude Information". In: *Physical Review E* 87.2 (Feb. 20, 2013), p. 022911. ISSN: 1539-3755, 1550-2376. DOI: 10.1103/PhysRevE.87.022911. URL: https://doi.org/10.1103/PhysRevE.87.022911 (visited on 04/11/2023).

[Fer+13] Basura Fernando et al. “Unsupervised visual domain adaptation using subspace alignment”. In: *Proceedings of the IEEE international conference on computer vision*. 2013, pp. 2960–2967.

[GH13] John Guckenheimer and Philip Holmes. *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*. Vol. 42. Springer Science & Business Media, 2013.

[Har+13] Sebastian de Haro et al. “Forty years of string theory reflecting on the foundations”. In: *Foundations of Physics* 43.1 (2013), pp. 1–7.

[LV13] M. Li and P. Vitanyi. *An Introduction to Kolmogorov Complexity and Its Applications*. Monographs in Computer Science. Springer New York, 2013. ISBN: 9781475738605. URL: https://books.google.pt/books?id=JvXiBwAAQBAJ.

[AGK14] KM Ahmmed, Colin Grambow, and Anne-Marie Kietzig. “Fabrication of Micro/nano Structures on Metals By Femtosecond Laser Micromachining”. In: *Micromachines* 5.4 (2014), pp. 1219–1253.

[Bév+14] E. Bévillon et al. “Free-electron properties of metals under ultrafast laser-induced electron-phonon nonequilibrium: A first-principles study”. In: *Phys. Rev. B* 89 (11 Mar. 2014), p. 115117.

[Bus14] F. H. Busse. “Remarks on the critical value Pc=0.25 of the Prandtl number for internally heated convection found by Tveitereid and Palm”. In: *Eur. J. Mech. B Fluids* 47 (Sept. 2014), pp. 32–34. ISSN: 0997-7546. DOI: 10.1016/j.euromechflu.2014.04.001.

[KSJ14] Angjoo Kanazawa, Abhishek Sharma, and David Jacobs. “Locally scale-invariant convolutional neural networks”. In: *arXiv preprint arXiv:1412.5104* (2014).

[NTS14] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. “In Search of the Real Inductive Bias: on the Role of Implicit Regularization in Deep Learning”. In: *arXiv preprint arXiv:1412.6614* (2014).

[Sri+14] Nitish Srivastava et al. “Dropout: a Simple Way To Prevent Neural Networks From Overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[Sun+14] Xiaoran Sun et al. “Characterizing System Dynamics With a Weighted and Directed Network Constructed From Time Series Data”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 24.2 (2014), p. 024402.

[VH14] Tim Van Erven and Peter Harremos. “Rényi divergence and Kullback-Leibler divergence”. In: *IEEE Transactions on Information Theory* 60.7 (2014), pp. 3797–3820.

[Cho+15] François Chollet et al. *Keras*. https://github.com/fchollet/keras. 2015.

[He+15] Kaiming He et al. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[HL15] David P Helmbold and Philip M Long. “On the Inductive Bias of Dropout”. In: *The Journal of Machine Learning Research* 16.1 (2015), pp. 3403–3454.

[Hu+15] Bing Hu et al. “Using the Minimum Description Length To Discover the Intrinsic Cardinality and Dimensionality of Time Series”. In: *Data Mining and Knowledge Discovery* 29 (2015), pp. 358–399.

[IS15] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *International conference on machine learning*. PMLR. 2015, pp. 448–456.

[KKP15] Anna Kalogirou, Eric E Keaveny, and Demetrios T Papageorgiou. "An in-depth numerical study of the two-dimensional Kuramoto–Sivashinsky equation". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 471.2179 (2015), p. 20140932.

[LV15] Karel Lenc and Andrea Vedaldi. "Understanding image representations by measuring their equivariance and equivalence". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 991–999.

[Li+15] Wanbo Li et al. "Well-Designed Metal Nanostructured Arrays for Label-Free Plasmonic Biosensing". In: *Journal of Materials Chemistry C* 3.25 (2015), pp. 6479–6492.

[Mar+15] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[SZ15] Karen Simonyan and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1409.1556.

[SP15] Sergey V. Starikov and Vasily V. Pisarev. "Atomistic simulation of laser-pulse surface modification: Predictions of models with various length and time scales". In: *J. Appl. Phys.* 117.13 (Apr. 2015), p. 135901. ISSN: 0021-8979. DOI: 10.1063/1.4916600.

[Sto+15] Norbert Stoop et al. "Curvature-induced symmetry breaking determines elastic surface patterns". In: *Nature materials* 14.3 (2015), pp. 337–342.

[Taf+15] Ahmad P Tafti et al. "Recent advances in 3D SEM surface reconstruction". In: *Micron* 78 (2015), pp. 54–66.

[TZ15] Naftali Tishby and Noga Zaslavsky. "Deep learning and the information bottleneck principle". In: *2015 IEEE Information Theory Workshop (ITW)*. IEEE. 2015, pp. 1–5.

[TFS15] George D. Tsibidis, C. Fotakis, and E. Stratakis. "From ripples to spikes: A hydrodynamical mechanism to interpret femtosecond laser-induced self-assembled structures". In: *Phys. Rev. B* 92.4 (July 2015), p. 041405. ISSN: 2469-9969. DOI: 10.1103/PhysRevB.92.041405.

[VG15] AY Vorobyev and Chunlei Guo. "Multifunctional Surfaces Produced By Femtosecond Laser Pulses". In: *Journal of Applied Physics* 117.3 (2015), p. 033103.

[Zha+15] Kai Zhao et al. "Explaining the Power-Law Distribution of Human Mobility Through Transportationmodality Decomposition". In: *Scientific reports* 5.1 (2015), pp. 1–7.

[ZOR15] Luciano Zunino, Felipe Olivares, and Osvaldo A Rosso. "Permutation Min-Entropy: An Improved Quantifier for Unveiling Subtle Temporal Correlations". In: *EPL (Europhysics Letters)* 109.1 (Jan. 1, 2015), p. 10005. ISSN: 0295-5075, 1286-4854. DOI: 10.1209/0295-5075/109/10005. URL: https://doi.org/10.1209/0295-5075/109/10005 (visited on 04/11/2023).

[Bon+16] Jörn Bonse et al. "Laser-Induced Periodic Surface Structures-A Scientific Evergreen". In: *IEEE Journal of selected topics in quantum electronics* 23.3 (2016).

[But16] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2016.

[CW16] Taco Cohen and Max Welling. "Group Equivariant Convolutional Networks". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 20–22 Jun 2016, pp. 2990–2999. URL: http://proceedings.mlr.press/v48/cohenc16.html.

[Cun+16] Alexandre Cunha et al. "Femtosecond Laser Surface Texturing of Titanium As a Method To Reduce the Adhesion of Staphylococcus Aureus and Biofilm Formation". In: *Applied Surface Science* 360 (2016), pp. 485–493.

[FMN16] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. "Testing the manifold hypothesis". In: *Journal of the American Mathematical Society* 29.4 (2016), pp. 983–1049.

[GJ16] Tianxiang Gao and Vladimir Jojic. "Degrees of Freedom in Deep Neural Networks". In: *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*. UAI'16. Jersey City, New Jersey, USA: AUAI Press, 2016, pp. 232–241. ISBN: 9780996643115.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[HRS16] Moritz Hardt, Ben Recht, and Yoram Singer. "Train faster, generalize better: Stability of stochastic gradient descent". In: *International conference on machine learning*. PMLR. 2016, pp. 1225–1234.

[He+16] 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.

[HG16] Dan Hendrycks and Kevin Gimpel. "Gaussian error linear units (gelus)". In: *arXiv preprint arXiv:1606.08415* (2016).

[Ram+16] Abdiel Ramírez-Reyes et al. "Determining the Entropic Index Q of Tsallis Entropy in Images Through Redundancy". In: *Entropy* 18.8 (2016), p. 299.

[Sil+16] David Silver et al. "Mastering the game of Go with deep neural networks and tree search". In: *nature* 529.7587 (2016), pp. 484–489.

[Sze+16] Christian Szegedy et al. "Rethinking the inception architecture for computer vision". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[Tsi+16] George D Tsibidis et al. "Convection roll-driven generation of supra-wavelength periodic surface structures on dielectrics upon irradiation with femtosecond pulsed lasers". In: *Physical Review B* 94.8 (2016), p. 081305.

[Fer+17] Max Ferguson et al. "Automatic localization of casting defects with convolutional neural networks". In: *2017 IEEE international conference on big data (big data)*. IEEE. 2017, pp. 1726–1735.

[Gni+17] Iaroslav Gnilitskyi et al. “High-Speed Manufacturing of Highly Regular Femtosecond Laser-Induced Periodic Surface Structures: Physical Origin of Regularity”. In: *Scientific reports* 7.1 (2017), pp. 1–11.

[Gua+17] Jean-Michel Guay et al. “Laser-induced plasmonic colours on metals”. In: *Nature communications* 8.1 (2017), pp. 1–12.

[Ild+17] Serim Ilday et al. “Rich complex behaviour of self-assembled nanoparticles far from equilibrium”. In: *Nature communications* 8.1 (2017), pp. 1–10.

[Meu+17] Aaron Meurer et al. “SymPy: symbolic computing in Python”. In: *PeerJ Computer Science* 3 (2017), e103. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.103. URL: https://doi.org/10.7717/peerj-cs.103.

[Rud+17] Samuel H Rudy et al. “Data-driven discovery of partial differential equations”. In: *Science advances* 3.4 (2017), e1602614.

[Sze+17] Christian Szegedy et al. “Inception-v4, inception-resnet and the impact of residual connections on learning”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 31. 1. 2017.

[TVV17] Vipin N. Tondare, John S. Villarrubia, and András E. Vladár. “Three-Dimensional (3D) Nanometrology Based on Scanning Electron Microscope (SEM) Stereophotogrammetry”. In: *Microscopy and Microanalysis* 23.5 (2017), pp. 967–977. DOI: 10.1017/S1431927617012521.

[YM17] Yuichi Yoshida and Takeru Miyato. “Spectral Norm Regularization for Improving the Generalizability of Deep Learning”. In: *arXiv preprint arXiv:1705.10941* (2017).

[Zha+17] Chiyuan Zhang et al. “Understanding deep learning requires rethinking generalization”. In: *International Conference on Learning Representations*. 2017. URL: https://openreview.net/forum?id=Sy8gdB9xx.

[Aro+18] Sanjeev Arora et al. “Stronger Generalization Bounds for Deep Nets Via a Compression Approach”. In: *CoRR* (2018). arXiv: 1802.05296 [cs.LG]. URL: http://arxiv.org/abs/1802.05296v4.

[Ben18] Charles H Bennett. “Dissipation, information, computational complexity and the definition of organization”. In: *Emerging syntheses in science*. Ed. by David Pines. CRC Press, 2018, pp. 215–233. ISBN: 9780429961144.

[BO18] Léonard Blier and Yann Ollivier. “The description length of deep learning models”. In: *Advances in Neural Information Processing Systems* 31 (2018).

[Car+18] Alberto Carrassi et al. “Data Assimilation in the Geosciences: an Overview of Methods, Issues, and Perspectives”. In: *Wiley Interdisciplinary Reviews: Climate Change* 9.5 (2018), e535.

[Che+18] Ricky TQ Chen et al. “Neural Ordinary Differential Equations”. In: *Advances in neural information processing systems* 31 (2018).

[Gei+18] Robert Geirhos et al. “ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness”. In: *arXiv preprint arXiv:1811.12231* (2018).

[Gol+18a] J.I. Goldstein et al. *Scanning Electron Microscopy and X-Ray Microanalysis*. Springer New York, 2018. ISBN: 9781493982691. URL: https://books.google.fr/books?id=uapMwwEACAAJ.

[Gol+18b] Joseph I Goldstein et al. "Electron beam—specimen interactions: Interaction volume". In: *Scanning Electron Microscopy and X-Ray Microanalysis* (2018), pp. 1–14.

[LLD18] Zichao Long, Yiping Lu, and Bin Dong. "PDE-Net 2.0: Learning PDEs from Data with A Numeric-Symbolic Hybrid Deep Network". In: *CoRR* abs/1812.04426 (2018). arXiv: 1812.04426. URL: http://arxiv.org/abs/1812.04426.

[Luo+18] Ping Luo et al. "Towards Understanding Regularization in Batch Normalization". In: *arXiv preprint arXiv:1809.00846* (2018).

[Lut+18] Adrian HA Lutey et al. "Towards Laser-Textured Antibacterial Surfaces". In: *Scientific reports* 8.1 (2018), pp. 1–10.

[Mar+18] Diego Marcos et al. "Scale equivariance in CNNs with vector fields". In: *arXiv preprint arXiv:1807.11783* (2018).

[MM18] Charles H Martin and Michael W Mahoney. "Implicit Self-Regularization in Deep Neural Networks: Evidence From Random Matrix Theory and Implications for Learning". In: *arXiv preprint arXiv:1810.01075* (2018).

[Mol+18] Sean Molesky et al. "Inverse Design in Nanophotonics". In: *Nature Photonics* 12.11 (2018), pp. 659–670.

[Mor+18a] Ari S. Morcos et al. "On the Importance of Single Directions for Generalization". In: *CoRR* (2018). arXiv: 1803.06959 [stat.ML]. URL: http://arxiv.org/abs/1803.06959v4.

[Mor+18b] R. V. Morgan et al. "Rarefaction-driven Rayleigh–Taylor instability. Part 2. Experiments and simulations in the nonlinear regime". In: *J. Fluid Mech.* 838 (Mar. 2018), pp. 320–355. ISSN: 0022-1120. DOI: 10.1017/jfm.2017.893.

[Qia+18] Hongzhen Qiao et al. "Formation of subwavelength periodic triangular arrays on tungsten through double-pulsed femtosecond laser irradiation". In: *Materials* 11.12 (2018), p. 2380.

[Rai18] Maziar Raissi. "Deep hidden physics models: Deep learning of nonlinear partial differential equations". In: *The Journal of Machine Learning Research* 19.1 (2018), pp. 932–955.

[San+18] Shibani Santurkar et al. "How Does Batch Normalization Help Optimization?" In: *arXiv preprint arXiv:1805.11604* (2018).

[SPR18] Higor YD Sigaki, Matjaž Perc, and Haroldo V Ribeiro. "History of Art Paintings Through the Lens of Entropy and Complexity". In: *Proceedings of the National Academy of Sciences* 115.37 (2018), E8585–E8594.

[Abo+19] Anthony Abou Saleh et al. "Nanoscale imaging of ultrafast light coupling to self-organized nanostructures". In: *ACS photonics* 6.9 (2019), pp. 2287–2294.

[Ara+19] Kerim Tugrul Arat et al. "Estimating step heights from top-down sem images". In: *Microscopy and Microanalysis* 25.4 (2019), pp. 903–911.

[Beu+19] Tom Beucler et al. "Achieving Conservation of Energy in Neural Network Emulators for Climate Modeling". In: *arXiv preprint arXiv:1906.06622* (2019).

[Bre+19] PG Breen et al. "Newton vs the machine: solving the chaotic three-body problem using deep neural networks. arXiv e-prints". In: *arXiv preprint arXiv:1910.07291* (2019).

[Che+19] Zhengdao Chen et al. "Symplectic Recurrent Neural Networks". In: *arXiv preprint arXiv:1909.13334* (2019).

[DM19] Alfonso Delgado-Bonal and Alexander Marshak. "Approximate Entropy and Sample Entropy: A Comprehensive Tutorial". In: *Entropy* 21.6 (May 28, 2019), p. 541. ISSN: 1099-4300. DOI: 10.3390/e21060541. URL: https://doi.org/10.3390/e21060541 (visited on 04/11/2023).

[Fra+19] Fotis Fraggelakis et al. "Controlling 2D laser nano structuring over large area with double femtosecond pulses". In: *Applied Surface Science* 470 (2019), pp. 677–686.

[GG19] Rohan Ghosh and Anupam K Gupta. "Scale Steerable Filters for Locally Scale-Invariant Convolutional Neural Networks". In: *arXiv preprint arXiv:1906.03861* (2019).

[GDY19] Samuel Greydanus, Misko Dzamba, and Jason Yosinski. "Hamiltonian Neural Networks". In: *Advances in Neural Information Processing Systems* 32 (2019).

[GR19] Peter Grünwald and Teemu Roos. "Minimum Description Length Revisited". In: *International Journal of Mathematics for Industry* 11.01 (2019), p. 1930001. DOI: 10.1142/S2661335219300018. eprint: https://doi.org/10.1142/S2661335219300018. URL: https://doi.org/10.1142/S2661335219300018.

[HS19] Hangfeng He and Weijie J Su. "The Local Elasticity of Neural Networks". In: *arXiv preprint arXiv:1910.06943* (2019).

[Jia+19] Yiding Jiang et al. "Fantastic Generalization Measures and Where to Find Them". In: *ICLR*. 2019.

[Min19] Paul Vitányi Ming Li. *An Introduction to Kolmogorov Complexity and Its Applications (Texts in Computer Science)*. 4th ed. 2019. Texts in Computer Science. Springer, 2019. ISBN: 3030112977,9783030112974. URL: http://gen.lib.rus.ec/book/index.php?md5=97624F0CB2F04FF0D895D36055568EDA.

[RPK19] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-Informed Neural Networks: a Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations". In: *Journal of Computational physics* 378 (2019), pp. 686–707.

[Red+19] Ievgen Redko et al. *Advances in Domain Adaptation Theory*. Elsevier, Aug. 2019. URL: https://hal.archives-ouvertes.fr/hal-02286281.

[Rud+19a] Anton Rudenko et al. "Amplification and regulation of periodic nanostructures in multipulse ultrashort laser-induced surface evolution by electromagnetic-hydrodynamic simulations". In: *Phys. Rev. B* 99 (23 June 2019), p. 235412.

[Rud+19b] Anton Rudenko et al. "Self-organization of surfaces on the nanoscale by topography-mediated selection of quasi-cylindrical and plasmonic waves". In: *Nanophotonics* 8.3 (2019), pp. 459–465.

[Sax+19] Andrew M Saxe et al. "On the Information Bottleneck Theory of Deep Learning". In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.12 (Dec. 2019), p. 124020. DOI: 10.1088/1742-5468/ab3985. URL: https://doi.org/10.1088/1742-5468/ab3985.

[WW19] Daniel E Worrall and Max Welling. "Deep scale-spaces: Equivariance over scale". In: *arXiv preprint arXiv:1905.11697* (2019).

[Zhu+19] Yinhao Zhu et al. "Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data". In: *Journal of Computational Physics* 394 (2019), pp. 56–81.

[Abo+20] Anthony Abou Saleh et al. "Sub-100 Nm 2d Nanopatterning on a Large Scale By Ultrafast Laser Energy Regulation". In: *Nanoscale* 12.12 (Mar. 2020), pp. 6609–6616. ISSN: 2040-3364. DOI: 10.1039/C9NR09625F.

[BG20] Jörn Bonse and Stephan Gräf. "Maxwell Meets Marangoni-A Review of Theories on Laser-Induced Periodic Surface Structures". In: *Laser & Photonics Reviews* 14.10 (2020), p. 2000215.

[BMT20] Adrian Bulat, Brais Martinez, and Georgios Tzimiropoulos. "Bats: Binary architecture search". In: *European Conference on Computer Vision*. Springer. 2020, pp. 309–325.

[Cra+20] Miles Cranmer et al. "Lagrangian Neural Networks". In: *arXiv preprint arXiv:2003.04630* (2020).

[Déc+20] Marie Déchelle et al. "Bridging Dynamical Models and Deep Networks to Solve Forward and Inverse Problems". In: *NeurIPS workshop on Interpretable Inductive Biases and Physically Structured Learning*. 2020.

[EC20] S Echeverría-Alar and MG Clerc. "Labyrinthine Patterns Transitions". In: *Physical Review Research* 2.4 (2020), p. 042036.

[Fil+20] Arthur Filoche et al. "Completing physics-based models by learning hidden dynamics through data assimilation". In: *NeurIPS 2020, workshop AI4Earth*. 2020.

[FT20] Olga Fuks and Hamdi A Tchelepi. "Limitations of physics informed machine learning for nonlinear two-phase transport in porous media". In: *Journal of Machine Learning for Modeling and Computing* 1.1 (2020).

[Grä20] Stephan Gräf. "Formation of laser-induced periodic surface structures on different materials: fundamentals, properties and applications". In: *Advanced Optical Technologies* 9.1-2 (2020), pp. 11–39.

[Ish+20] Takashi Ishida et al. "Do we need zero training loss after achieving zero training error?" In: *arXiv preprint arXiv:2002.08709* (2020).

[MM20] Charles H Martin and Michael W Mahoney. "Heavy-tailed Universality predicts trends in test accuracies for very large pre-trained deep neural networks". In: *Proceedings of the 2020 SIAM International Conference on Data Mining*. SIAM. 2020, pp. 505–513.

[Ngu+20] Duong Nguyen et al. "Assimilation-based learning of chaotic dynamical systems from noisy and partial data". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 3862–3866.

[OMY20] Adam C Overvig, Stephanie C Malek, and Nanfang Yu. "Multifunctional nonlocal metasurfaces". In: *Physical Review Letters* 125.1 (2020), p. 017402.

[Rud+20] Anton Rudenko et al. "High-Frequency Periodic Patterns Driven By Non-Radiative Fields Coupled With Marangoni Convection Instabilities on Laser-Excited Metal Surfaces". In: *Acta Materialia* 194 (2020), pp. 93–105.

[Sch20] Peter Schweizer. "Gezielte Manipulation einzelner Defekte in Schichtkristallen und 2D-Materialien". PhD thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 2020.

[SC20] Razvan Stoian and Jean-Philippe Colombier. "Advances in ultrafast laser structuring of materials at the nanoscale". In: *Nanophotonics* 9.16 (2020), pp. 4665–4688.

[Str+20] E Stratakis et al. "Laser engineering of biomimetic surfaces". In: *Materials Science and Engineering: R: Reports* 141 (2020), p. 100562.

[Um+20] Kiwon Um et al. "Solver-in-the-Loop: Learning from Differentiable Physics to Interact with Iterative PDE-Solvers". In: *Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*. 2020.

[Vir+20] Pauli Virtanen et al. "SciPy 1.0: fundamental algorithms for scientific computing in Python". In: *Nature methods* 17.3 (2020), pp. 261–272.

[Vit+20] Eduardo Vitral et al. "Spiral defect chaos in Rayleigh-Bénard convection: Asymptotic and numerical studies of azimuthal flows induced by rotating spirals". In: *Phys. Rev. Fluids* 5.9 (Sept. 2020), p. 093501. ISSN: 2469-990X. DOI: 10.1103/PhysRevFluids.5.093501.

[Wan+20] Shaojun Wang et al. "Controllable formation of laser-induced periodic surface structures on ZnO film by temporally shaped femtosecond laser scanning". In: *Opt. Lett.* 45.8 (Apr. 2020), pp. 2411–2414. ISSN: 1539-4794. DOI: 10.1364/OL.388770.

[Wil+20] Jared Willard et al. "Integrating physics-based modeling with machine learning: A survey". In: *arXiv preprint arXiv:2003.04919* 1.1 (2020), pp. 1–34.

[Baa+21] Pieter Baatsen et al. "Preservation of fluorescence signal and imaging optimization for integrated light and electron microscopy". In: *Frontiers in Cell and Developmental Biology* 9 (2021), p. 737621.

[Beu+21] Tom Beucler et al. "Climate-Invariant Machine Learning". In: *arXiv preprint arXiv:2112.08440* (2021).

[Bra+21] Julien Brajard et al. "Combining Data Assimilation and Machine Learning To Infer Unresolved Scale Parametrization". In: *Philosophical Transactions of the Royal Society A* 379.2194 (2021), p. 20200086.

[Far+21] Alban Farchi et al. "Using Machine Learning To Correct Model Error in Data Assimilation and Forecast Applications". In: *Quarterly Journal of the Royal Meteorological Society* 147.739 (2021), pp. 3067–3084.

[Gra+21] Mara Graziani et al. "On the Scale Invariance in State of the Art Cnns Trained on Imagenet". In: *Machine Learning and Knowledge Extraction* 3.2 (2021), pp. 374–391. ISSN: 2504-4990. DOI: 10.3390/make3020019. URL: https://www.mdpi.com/2504-4990/3/2/19.

[Jia+21] Xiaowei Jia et al. "Physics-Guided Machine Learning for Scientific Discovery: an Application in Simulating Lake Temperature Profiles". In: *ACM/IMS Transactions on Data Science* 2.3 (2021), pp. 1–26.

[Kar+21] George Em Karniadakis et al. "Physics-Informed Machine Learning". In: *Nature Reviews Physics* 3.6 (2021), pp. 422–440.

[Ma+21] Wei Ma et al. "Deep Learning for the Design of Photonic Structures". In: *Nature Photonics* 15.2 (2021), pp. 77–90.

[Mas+21] Matteo Mastellone et al. "Deep-subwavelength 2D periodic surface nanostructures on diamond by double-pulse femtosecond laser irradiation". In: *Nano Letters* 21.10 (2021), pp. 4477–4483.

[Nak+21a] Anthony Nakhoul et al. "Self-Organization Regimes Induced By Ultrafast Laser on Surfaces in the Tens of Nanometer Scales". In: *Nanomaterials* 11.4 (Apr. 2021), p. 1020. ISSN: 2079-4991. DOI: 10.3390/nano11041020.

[Nak+21b] Preetum Nakkiran et al. "Deep double descent: Where bigger models and more data hurt". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12 (2021), p. 124003.

[VS21] Felipe AC Viana and Arun K Subramaniyan. "A Survey of Bayesian Calibration and Physics-Informed Neural Networks in Scientific Modeling". In: *Archives of Computational Methods in Engineering* 28.5 (2021), pp. 3801–3830.

[WTP21] Sifan Wang, Yujun Teng, and Paris Perdikaris. "Understanding and mitigating gradient flow pathologies in physics-informed neural networks". In: *SIAM Journal on Scientific Computing* 43.5 (2021), A3055–A3081.

[Wie+21] Peter R Wiecha et al. "Deep Learning in Nano-Photonics: Inverse Design and Beyond". In: *Photonics Research* 9.5 (2021), B182–B200.

[Xu21] Hao Xu. "DL-PDE: Deep-Learning Based Data-Driven Discovery of Partial Differential Equations from Discrete and Noisy Data". In: *Communications in Computational Physics* 29.3 (June 2021), pp. 698–728. DOI: 10.4208/cicp.oa-2020-0142.

[Yin+21] Yuan Yin et al. "Augmenting Physical Models With Deep Networks for Complex Dynamics Forecasting". In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12 (2021), p. 124012.

[YPK21] Rose Yu, Paris Perdikaris, and Anuj Karpatne. "Physics-Guided AI for Large-Scale Spatiotemporal Data". In: *The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 2021, pp. 4088–4089.

[Zha+21] Chiyuan Zhang et al. "Understanding Deep Learning (still) Requires Rethinking Generalization". In: *Communications of the ACM* 64.3 (2021), pp. 107–115.

[Bra+22] Eduardo Brandao et al. "Learning Pde To Model Self-Organization of Matter". In: *Entropy* 24.8 (2022), p. 1096. ISSN: 1099-4300. DOI: 10.3390/e24081096. URL: https://www.mdpi.com/1099-4300/24/8/1096.

[DNT22] Krish Desai, Benjamin Nachman, and Jesse Thaler. "Symmetry discovery with deep learning". In: *Physical Review D* 105.9 (2022), p. 096031.

[Don+22] Jérémie Dona et al. "Constrained Physical-Statistics Models for Dynamical System Identification and Prediction". In: *International Conference on Learning Representations (ICLR)*. 2022.

[Gal22] Esther Galbrun. "The Minimum Description Length Principle for Pattern Mining: a Survey". In: *Data mining and knowledge discovery* 36.5 (2022), pp. 1679–1727.

[Hou+22] Tim Houben et al. "Depth estimation from a single SEM image using pixel-wise fine-tuning with multimodal data". In: *Machine Vision and Applications* 33.4 (2022), p. 56.

[Ley+22] Inmaculada Leyva et al. “20 Years of Ordinal Patterns: Perspectives and Challenges”. In: *Europhysics Letters* 138.3 (2022), p. 31001. DOI: 10.1209/0295-5075/ac6a72. URL: https://doi.org/10.1209/0295-5075/ac6a72.

[Nak+22] Anthony Nakhoul et al. “Boosted Spontaneous Formation of High-Aspect Ratio Nanopeaks on Ultrafast Laser-Irradiated Ni Surface”. In: *Advanced Science* (2022), p. 2200761.

[Son+22] Hwanjun Song et al. “Learning From Noisy Labels With Deep Neural Networks: a Survey”. In: *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[WYP22] Sifan Wang, Xinling Yu, and Paris Perdikaris. “When and why PINNs fail to train: A neural tangent kernel perspective”. In: *Journal of Computational Physics* 449 (2022), p. 110768.

[Wei+22] Jiaheng Wei et al. “Learning with Noisy Labels Revisited: A Study Using Real-World Human Annotations”. In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=TBWA6PLJZQm.

[Bra+23] Eduardo Brandao et al. “Learning Complexity to Guide Light-Induced Self-Organized Nanopatterns”. In: *Physical Review Letters* 130.22 (2023), p. 226201.

[Via+23] Paul Viallard et al. “Generalization Bounds with Arbitrary Complexity Measures”. In: *Submitted to The Eleventh International Conference on Learning Representations*. under review. 2023. URL: https://openreview.net/forum?id=WhwtdGkbaDr.

[Li+] Chunyuan Li et al. “Measuring the Intrinsic Dimension of Objective Landscapes”. In: *International Conference on Learning Representations*.