

Foretelling Microstructural Interface with Multi-Generational Convolutional-LSTM Framework

Upadesh Subedi^{1*}, Nele Moelans², Tomasz Tański¹, Anil Kunwar¹

¹Faculty of Mechanical Engineering, Silesian University of Technology,
Konarskiego 18A, Gliwice, 44-100, Poland.

²Department of Materials Engineering, KU Leuven, Kasteelpark
Arenberg 44 Bus 2450, Leuven, 3001, Belgium.

*Corresponding author(s). E-mail(s): upadesh.subedi@polsl.pl;

Abstract

Predicting multi-generational microstructural evolution in the manifest space using Convolutional-LSTM neural network models is a promising approach towards computational reducibility. Two predictive models were trained, analyzed, and compared using data generated from varying the driving force in phase-field simulations of phase decomposition in a binary alloy system. The four consecutive predicted generations exhibited microstructural similarity indices (μ_{SIM}) of 0.994, 0.987, 0.980, and 0.967, respectively, compared to the ground truth computational results. By utilizing blobs, μ_{SIM} , and shape index features, we validated that the predicted images retain microstructural information down to the interfacial level. The time-aware data-driven model was applied to rapidly forecast structural changes in Pd-Rh and Cu-Mn engineering alloys.

Keywords: Conv-LSTM, Shape Index, Double-well potential, Phase Field Modeling, Microstructural Similarity, Multi Generational Prediction

1 Introduction

The microstructural evolution of any material fundamentally dictates its physical and mechanical properties observed at the macro-level. Therefore, the spatio-temporal study of microstructure stands as a paramount requirement for comprehending the

design and development of new materials [1]. Understanding a material's morphological transformations can directly facilitate the discovery of concealed microstructural imperfections, which, in sensitive cases, can lead to possible catastrophic failures [2–7]. In such scenarios, the phase-field method takes the forefront in mesoscale microstructural studies, enabling the simulation of phenomena such as grain growth [8], crack propagation [9, 10], spinodal decomposition [11], and coarsening [12]. Surpassing traditional methods in material design and discovery, the new era of Industry 4.0 is currently witnessing an upswing, thanks to state-of-the-art in-silico data-driven approaches. In this fast-paced and rapidly evolving technological landscape, leveraging big data informatics to decipher complex microstructural evolution is imperative to stay ahead of innovations in materials [13, 14].

The phase-field method is undeniably a robust, multifaceted, and versatile technique for microstructural simulation. However, it consumes an astronomical amount of computational resources to solve the system of nonlinear coupled partial differential equations (PDEs). This presents a significant barrier to achieving computational reducibility. At every step of microstructural evolution study, phase-field simulation requires solving complex PDEs with intricate physics and geometry, posing challenges in computation and convergence criteria.

1.1 Background

A sound understanding and proficiency in microstructural evolution are critical for materials utilized in essential infrastructures, such as nuclear reactors, space launch systems, high-performance locomotives, and life-saving devices like artificial pacemakers or stents in blood vessels. Even minor microstructural imperfections in these devices can lead to significant property damage or loss of human life. Therefore, a scientifically sound and physics-backed approach to spatio-temporal prediction of microstructure is necessary. Proper and accurate inference of microstructural evolution enables the timely detection of critical failure points in high-impact mechanical systems within infrastructure. Phenomena such as the corrosion of alloys in molten salt reactors [15], high-temperature crack growth in turbine blades [16], or mechanical strength deterioration in biomedical implants due to phase transformation [17, 18] can all be assessed and mitigated early with the aid of physics-informed microstructure predictive models.

1.2 Existing studies

Numerous researchers have proposed various methods and models to analyze microstructural patterns and their temporal transformations. However, the majority of these approaches lack either adequate situational awareness or a robust foundation of physics-informed guidance. Furthermore, almost none of the studies have primarily focused on anticipating interface-focused microstructure evolution.

Recently, Machine Learning (ML) models, particularly Artificial Neural Networks (ANNs), have emerged as leading tools in predictive modeling. Time-aware neural networks such as Recurrent Neural Networks (RNNs), Long Short-Term Memory

(LSTM), and Gated Recurrent Units (GRU) have shown promising results in predicting microstructural evolution to some extent [19, 20]. However, most of these studies that solely rely on standalone vanilla neural network algorithms are insufficient in predicting quantifiable microstructural evolution. Additionally, they often lack a thermodynamically consistent calculation of driving force (free energy).

Hu et al. [21] employed simple RNN, GRU, and vanilla LSTM models as microstructural predictive tools in their work. Nevertheless, inferring quantitative parameters from their model proved challenging as the predictions primarily reside in a 'latent space'. Predictions based on latent space offer computational ease, but the transformation of high-dimensional spatial data into a reduced form results in the loss of essential spatial information present in the microstructure. Consequently, any correlations between neighboring spatial coordinates of microstructures are lost as they are diminished and flattened into an array of numbers.

Furthermore, it is important to note that the predictions made within these latent spaces are not readily understandable to end-users and require additional computational steps to convert them into human-interpretable 2D image vectors. Moreover, the lack of quantification measures in these works exacerbates the challenge of dimensionality. The robustness of any machine learning model is determined by the quantification metrics for its predicted results; a higher numerical resemblance of predicted results to that of the ground truth indicates a reliable and better-performing model. A work by Farizhandi et al. [22] uses PredRNN to predict microstructure changes in the FeCrCo system, but their study lacks a comprehensive analysis of generational prediction behavior across different model architectures, and the limited range of quantification techniques employed further undermines its reliability. Ahmad et al. [23], in their study, cleverly compressed microstructural data to enhance computational efficiency, representing the spatial information of microstructures in their nascent form using auto-encoders and ConvLSTM. However, their work lacks error quantification at the interface level of the predicted microstructure. In the phase-field method, interface informatics play a crucial role, as the method itself is a numerical approach to simulate and track the microstructural evolution of phase boundaries and interfaces over time. Quantifying microstructural morphology at interfaces provides valuable insights into understanding the sensitivity of the phase-field model and its key parameters controlling microstructural evolution.

1.3 Computer-learned interface pattern : the solution

To address the challenges related to prediction, quantification, and model application, we propose a spatio-temporal multi-generational prediction toolkit for microstructures, employing Convolutional LSTM. This toolkit aims to proficiently decode the spatio-temporal progression of microstructural interfaces by intricately combining two cutting-edge machine learning models: Convolutional layers and LSTM neural networks. By integrating these disruptive tools, our toolkit strives to achieve computational reducibility in the analysis of microstructural evolution.

Conv-LSTM offers a two-fold solution to the major challenges faced in contemporary microstructural prediction methods. Firstly, it eliminates the need for dimensional reduction detours by enabling the direct implementation of high-dimensional spatial

microstructural data into the model, thereby providing a more comprehensive representation of microstructure within the prediction process. Secondly, and perhaps most importantly, Conv-LSTM facilitates the prediction of microstructural images in manifest spaces, rather than latent spaces, enabling direct quantitative analysis of predicted microstructures. Additionally, by leveraging one of the key features of Conv-LSTM, its ability to incorporate multi-dimensional, multi-channel input, we can expedite predictions across multiple future time steps, making it a multi-generational prediction tool.

In Section 3, it is explained in detail about how we employed specialized algorithms to track, detect, and differentiate interfacial patterns in microstructures, quantifying the comparison between the model-predicted and simulation-generated results.

2 Method

To prognosticate the spatio-temporal microstructural evolution in a generic binary (A-B) alloy system, driven by Gibbs Free energy (ranging from -10^4 J/mol), we synergistically combined two state-of-the-art, trailblazing Artificial Intelligence Algorithms: Convolutional Neural Networks (CNNs) and Long Short Term Memory (LSTM) neural networks. The integration of these two algorithms resulted in our functional prediction model, Conv-LSTM.

Our study challenges the practice of converting high-resolution two-dimensional simulation results into a non-user-friendly latent space for dimensional reduction and microstructure prediction. We argue that such transformations are unproductive and cumbersome, leading to information loss. The transformation process diminishes resolution by reducing the information present in high-dimensional microstructure data. Moreover, since prediction also occurs within the latent space, hindering user understandability, it undermines and discards the very essence of spatial data. Lastly, the retrieval of predicted outcomes from a low-dimensional latent space shaped back as a user-understandable image vector becomes a daunting task, necessitating secondary models.

Harnessing the capabilities of Convolutional Layers, we efficiently address the inefficiencies associated with conventional approaches involving latent spaces. These layers enable the processing of two-dimensional microstructural image vectors through various vector manipulation techniques such as resizing, pooling, and normalization, before passing the microstructural data to the LSTM neural network. The integration of the LSTM network on top of Convolutional Layers eliminates the need for dimensional transformation through latent space, allowing direct time-aware prediction of microstructural evolution in high-dimensional image format. This novel approach not only enhances the accuracy and usability of the results but also ensures the preservation of spatial information and the co-relationship between adjacent microstructural patterns.

Additionally, it is essential to highlight that the significant advantage of utilizing Conv-LSTM extends beyond generational microstructural prediction in high-definition image format. One of the greatest benefits lies in the ability to leverage a wide range of image quantification techniques derived from computer vision research.

These techniques allow us to visually and quantitatively assess the machine-predicted microstructural progression. In our study, we have considered these image quantification tools, including the Microstructural Similarity Index (μ_{SIM}), gray image histogram for pixel intensity, microstructural blob detection, and shape curvature index. By incorporating such quantification methods, we establish our work as a benchmark model for robust microstructural prediction, providing a comprehensive and rigorous analysis of multi-generational predicted results.

2.1 Phase Field Simulation and Spatio-Temporal Dataset

Multiple phase field simulations were conducted for a generic (A_i - B_i) binary alloy system undergoing phase decomposition driven by the driving force. Efforts were made to ensure that its free energies (G_i) fall within the range applicable to real engineering alloys.

The Cahn-Hilliard Equation 1 was solved using the finite element method within the MOOSE framework [24–27] to simulate the decomposition of the alloy system. The simulations were carried out on a 15 nm × 15 nm 2D mesh using QUAD4 elements.

$$\frac{\partial c}{\partial t} = \nabla \cdot M \nabla \left(\frac{\partial g(c)}{\partial c} - \kappa \nabla^2 c \right) \quad (1)$$

Here, 'c' represents the mole fraction of one of the elements in the binary alloy system, 'M' denotes the mobility of the same element with units of m^5/Js , and ' κ ' stands for the gradient energy coefficient with units of J/m . The key term in the Cahn-Hilliard equation is 'g(c)', which represents the free energy density of the system. Free energy plays a crucial role in driving the phase separation process by promoting the growth of compositional fluctuations during phase decomposition. The numerical value of free energy density can be calculated using two methods: (i) the system's Gibbs free energy (G), and (ii) the molar volume (V_{mol}) of the focused element, as shown in Equation 2.

$$g(c) = \frac{G}{V_{mol}} \quad (2)$$

In a binary alloy system, phase decomposition is governed by the minimization of Gibbs free energy, leading the system to evolve through compositional fluctuations to reach equilibrium at the lowest state of free energy. The Gibbs free energy curve for the phase decomposition of a binary alloy exhibits a characteristic double-well structure with two minima, each representing one of the two distinct coexisting phases. These minima correspond to the compositions of the separated phases in a stable equilibrium state, minimizing the free energy of the entire system. Various expressions for double-well free energy can be found in the literature [28–37] to model phase decomposition or spinodal decomposition. We introduce the following machine learning descriptors generating double-well function summarizing all of these variants to create a large set of microstructural evolution data and subsequently train the machine learning model.

$$G = \alpha(\epsilon c - c_{eq})^2 + \beta(\epsilon c - c_{eq}) + \gamma(\epsilon c - c_{eq})^4 + \delta \quad (3)$$

While quantitative polynomial equations of different orders [38] could be used to generalize the double-well free energy expressions, this study confines itself to a 4th order Equation 3 due to its established proximity with known thermodynamic free energy models such as Ginzburg-Landau and Flory-Huggins potentials [36, 39, 40]. This expression for the driving force (G [J/mol]) comprises six different coefficients ($\alpha, \beta, \gamma, \delta, \epsilon$, and c_{eq}) that influence the shape and structure of the free energy curve. The flexibility of this polynomial allows for a wide variance in the Gibbs free energy landscape, which is crucial for generating a diverse dataset for training the neural network model.

In this expression, the coefficients α, β, γ , and δ share the same unit as Gibbs free energy, i.e., J/mol , while ϵ and c_{eq} are unitless parameters. The numerical range for each of these features (coefficients) is provided in Table 1, and these variations are purposefully made to ensure a diverse training dataset obtained from microstructure phase-field simulations. The impact of variations in these six coefficients and parameters on the Gibbs free energy landscape is discussed in detail in Appendix A. To ensure that G attains realistic values, the coefficient δ is approximated to have a magnitude closer to the global minima of the Au-Pt alloy system at 1525.15 K. In addition to varying the Gibbs free energy attributes, features corresponding to fluctuations in the model were implemented via the initial seed number of nuclei, as shown in Figure 1b.

Table 1 Coefficients of free energy expression 3 used as features for creation of variable phase field simulations so as to create diverse range of training data. * The minimum and maximum values of these variables are statistically assigned around the central values based upon the CALPHAD information of the global minima for free energy curve of standard Au-Pt binary alloy at 1525.15K

Features	Minimum Value*	Maximum Value*	Unit
α	-8.9128e4	-4.3879e4	(J/mol)
β	-2.0e3	3.0e3	(J/mol)
γ	1.15e5	7.0e5	(J/mol)
δ	-5.736e3	-4.604e3	(J/mol)
ϵ	1.2	1.9	-
c_{eq}	0.5	0.7476	-

Using the expression for thermodynamically consistent and physically realistic Gibbs free energy, we conducted simulations with different landscapes of the driving force. The mobility (M) and gradient energy coefficient (κ) were held constant at $2.329 \times 10^{-31} m^5/J s$ and $7.965 \times 10^{-8} J/m$ for all simulations. In Section 3.3, we will discuss how the kinetics of the reaction can be virtually varied by sequencing images in model training. Leveraging this concept of virtually varying kinetic parameters, we performed phase-field simulations while keeping mobility constant. Figure 1a illustrates the Gibbs free energy curve for one of the simulations conducted at a temperature of 1525.15 K. The simulation results were visualized using ParaView, a

visualization toolkit [41]. The composition tensor at every mesh coordinate for each time step was extracted into a CSV file. These composition tensors were then used to create image-vector frames (F_s) measuring 401×401 pixels. For computational efficiency, all image vectors were reshaped to 64×64 pixels, as shown in Figure 1b.

To train the neural network model, we constructed a large set of video samples (VS) by grouping 20 frames of image vectors into a 3D image tensor, where each successive image vector was δ frames apart. Each video sequence consisted of exactly 20 frames. To curate the training-validation dataset, a dedicated algorithm was developed to generate video sequences from image-vector frames obtained from six phase-field simulations, as depicted in Figure 1c. In total, 1196 image-vector frames were generated from the six simulations, and 3625 different video sequences were constructed using the algorithm.

This process ensured that our dataset contained a diverse set of microstructural evolution scenarios, enabling the Conv-LSTM model to learn and predict the temporal progression of microstructural features effectively. The generated video sequences not only captured a broad range of phase decomposition phenomena but also provided the necessary data for training and validating the predictive model, as outlined in Section 3.3. For the training, validation, and testing purposes, 80% of the video data sequences were used for model training, while 10% of the dataset was reserved for validation and the remaining 10% for testing the trained model. The separation of data into these subsets was done to ensure a robust evaluation of the model's performance across unseen data, thereby assessing its generalizability and accuracy in predicting microstructural evolution.

2.2 CNN and LSTM models

Neural networks are complex systems of mathematical and statistical computational steps designed to approximate functions using data points. Convolutional Neural Networks (CNNs) represent a groundbreaking advancement in machine learning, initially introduced by LeCun et al. [43] in 1995. CNNs are specifically designed to handle high-dimensional data, such as images or video frames, without requiring feature reduction techniques like latent spaces or principal component analysis. In this study, we utilized CNNs to effectively process phase-field simulation-generated image-vector frames, allowing us to directly train the LSTM model.

Figure 2a illustrates the architecture of a three-layered convolutional network used in our approach. The network begins with a 5×5 kernel, progresses to a 3×3 kernel in the middle layer, and concludes with a 1×1 kernel in the final layer. Each convolutional layer applies its kernel across the entire image to extract spatial features, such as blobs, voids, or edges, which are then passed to the subsequent layer.

The choice of kernel sizes is crucial for the network's performance. A larger kernel size, such as 5×5 , is adept at capturing global features and patterns in high-dimensional data but increases computational time due to a higher number of associated parameters and may degrade spatial resolution in deeper layers. To balance feature extraction and computational efficiency, a medium-sized kernel (3×3) was used in the second layer. This kernel size captures important spatial details while

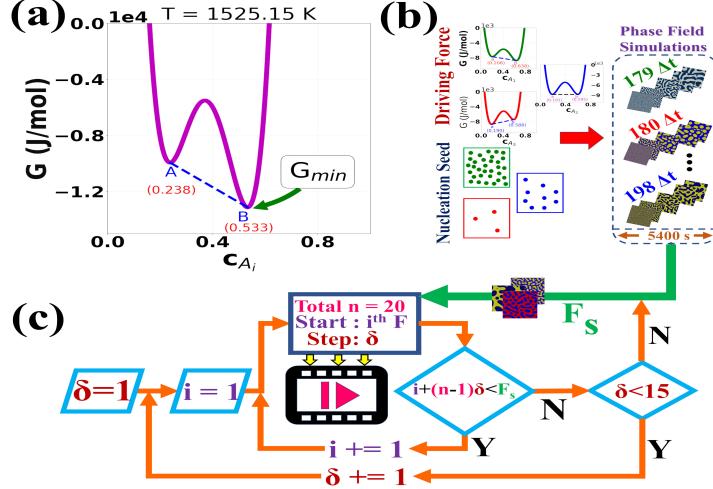


Fig. 1 Illustration of the phase decomposition in a generic A-B binary alloy system governed by the Gibbs Free Energy driving force. a) The thermodynamically consistent driving force curve fitting was performed at a temperature of 1525.15 K using composition-based TDB files [42] for one of the simulations in the *pycalphad* Python library. The curve exhibits two local minima, denoted as compositions A (0.238) and B (0.533), with the global minimum at B, having a value of $-1.31 \times 10^4 \text{ J/mol}$. b) A schematic diagram of the Phase Field simulations conducted in the MOOSE Framework [26]. Simulations varied in driving force and nucleation seed and ran for 5400 seconds each. The adaptive time iteration for convergence resulted in different time steps (Δt) for each simulation. Image vector data was extracted from each simulation using ParaView, treating each time step Δt simulation result as Image Vectors (F_s). c) Image vectors (F_s) were used to create a large set of video sequences through a custom-developed algorithm. Each video sequence consists of 20 image frames (n), curated by changing the starting frame ($i^{th} F$) and the step to the next frame (δ). Increasing the step size can be considered equivalent to increasing kinetics, making the reaction faster. Step sizes ranging from 1 to 15 were considered for the study.

maintaining reasonable computational demands and minimizing the risk of overfitting by limiting the number of parameters. The final layer employs a 1×1 kernel, which reduces the depth of the image channels, optimizing computational efficiency and memory usage.

Following the convolutional layers, the Long Short Term Memory (LSTM) network is applied to capture temporal dependencies and predict future microstructural states. LSTM networks, introduced by Hochreiter et al. [44], represents a significant advancement over the traditional Recurrent Neural Network (RNN). LSTMs were specifically developed to address the challenges associated with long-term dependencies in sequential data, such as the vanishing and exploding gradient problems that commonly affect standard RNNs [45]. LSTMs have since become a cornerstone in time series forecasting [46] and various other applications, including speech recognition and natural language processing [47].

At the heart of the LSTM architecture is the cell state or memory cell, which serves as a conduit for information flow across time steps in the sequence. Information flow in a cell state is regulated by a series of mechanisms referred as "gates." The input

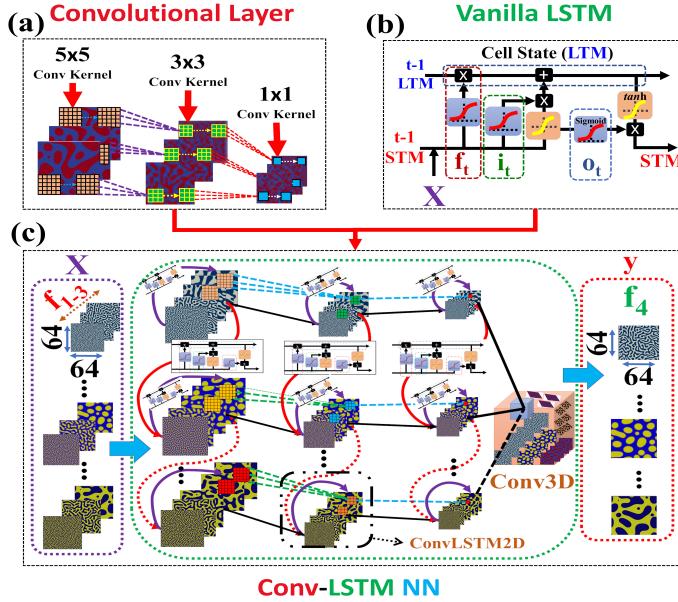


Fig. 2 Illustration of the components of the Convolutional LSTM Neural Network a) Three convolutional layers with kernels of sizes 5×5 , 3×3 , and 1×1 , respectively, process frames from Video Samples (VS). These layers analyze microstructures by convolving kernels around each pixel of the frame and pass condensed information from one layer to the next for further analysis. b) A schematic diagram of a vanilla LSTM Neural Network capable of intercepting Long-Term Memory (LTM) or Cell State and Short-Term Memory (STM) from the previous time period ($t-1$). It combines STM and Input (I/P) at time t , analyzing them through three gates: forget gate (f_t), input gate (i_t), and output gate (o_t). These gates control the amount of information retained and passed to the next time step in the form of LTM or STM using mathematical expressions defined in Equation 4. c) A complex model architecture featuring intertwined layers of Convolutional and LSTM networks, forming a unique ConvLSTM Neural Network structure. This work's model architecture takes three frames (f_{1-3}) of size 64×64 pixels from VS as input and predicts the fourth frame (f_4) in succession with the same pixel dimensions as output. The model can handle batches of VS simultaneously, predicting corresponding batches of the next frame, creating a 3D layer of convolved predictions.

gate determines the proportion of new input to be passed to the cell state, while the forget gate acts as a filter for information from the previous state. The output gate controls the information to carry from the current cell state to the final state. This stepwise calculation, filtering, and passing of information from the previous cell state to the next is the essence of LSTM.

Figure 2b illustrates the structure of a vanilla LSTM network and its gated mechanisms. The numerical expressions governing each gate in the network are represented in Equation 4 [45], where f_t , i_t , c_t , o_t , and h_t denote the input gate, forget gate, cell state, output gate, and final state, respectively, at time step t . The weights and biases of individual gates and states are represented by W and B with subscripts indicating the gate and time step. Sigmoid (σ) and tanh functions are employed as activation functions within these gates, as indicated in Figure 2b with red and yellow curves representing different gates.

A LSTM layer commences with the forget gate f_t (depicted as a red dotted rectangular box in Figure 2b). The sigmoid function is applied to the current input X_t and the Short-Term Memory from the previous time step STM_{t-1} , yielding a value between 0 and 1. This value is then used to pointwise multiply with the Long-Term Memory (LTM_{t-1}) to determine how much information from LTM should be discarded. Subsequently, LTM is updated by adding a portion of the potential long-term memory generated from the input gate (represented by the green dotted rectangle in Figure 2b). This update occurs in several steps: first, two operations are performed using X_t and STM_{t-1} , where information is extracted via the tanh function and a percentage value is determined with the sigmoid function. In the second step, the outcomes of these two operations are multiplied together, resulting in the amount of information added to LTM. The final stage of the LSTM network involves updating the short-term memory unit, where a sigmoid function is applied at the output gate (blue dotted box in Figure 2b) to X_t and STM_{t-1} , and this result is then multiplied by the outcome of the tanh function on LTM_t . The superscript α in Equation 4 indicates that these expressions are for the Vanilla LSTM network, and \circ represents the pointwise product.

$$\begin{aligned}
f_t^\alpha &= \sigma(W_f^\alpha [STM_{t-1}^\alpha, X_t^\alpha] + b_f^\alpha) \\
i_t^\alpha &= [\sigma(W_i^\alpha [STM_{t-1}^\alpha, X_t^\alpha] + b_i^\alpha)) \\
&\quad \circ (\tanh(W_i^\alpha [STM_{t-1}^\alpha, X_t^\alpha] + b_i^\alpha))] \\
LTM_t^\alpha &= f_t^\alpha \circ LTM_{t-1}^\alpha + i_t^\alpha \\
o_t^\alpha &= \sigma(W_o^\alpha [STM_{t-1}^\alpha, X_t^\alpha] + b_o^\alpha) \\
STM_t^\alpha &= o_t^\alpha \circ \tanh LTM_t
\end{aligned} \tag{4}$$

2.3 ConvLSTM NN

As discussed in Section 2.2, Convolutional Neural Networks (CNNs) and vanilla Long Short Term Memory (LSTM) networks are among the most powerful machine learning tools available, each serving distinct purposes when used individually. CNNs are commonly employed for image data processing tasks such as image classification and disease detection from medical images [48, 49]. However, they lack the inherent capability to handle temporal information in image data. In contrast, LSTM networks are designed for predicting time-dependent phenomena, such as time series forecasting and voice assistance. Despite this, vanilla LSTMs can only process data as one-dimensional vectors, which necessitates the flattening of any spatial information within input image-vectors before training.

To combine the strengths of these two algorithms, Shi et al. [50] introduced a fusion of Convolutional layers with LSTM neural networks, resulting in the Convolutional-LSTM (ConvLSTM) Neural Network. The equations used to describe this hybrid ConvLSTM network are provided in Equation 5. In this equation, X_t , C_t , and H_t represent the inputs, cell output (Long-Term Memory), and hidden states (Short-Term Memory) at time t , while i_t , f_t , and o_t denote the input, forget, and output gates, respectively. The symbol \odot represents the convolutional operator, and \circ signifies element-wise multiplication. Weights and biases for operations in each state and gate

are denoted by W and b , with lowercase subscripts indicating the respective state or gate. The superscript β in Equation 5 specifically denotes the Convolutional-LSTM network.

In simpler terms, it can be stated that with the advent of modern algorithms like transformers [51, 52], a plain vanilla LSTM model has become somewhat outdated. Studies that employ RNNs or standalone LSTMs [21], using latent space feature reduction for phase-field prediction, also face the limitation of not being able to process high-dimensional spatial image-vector data as input into the network. In this work, we address this significant issue by implementing the ConvLSTM NN and successfully predict the evolution of microstructures by training the model with spatial information of microstructures at different time steps in the form of image-vectors generated from phase field simulations.

$$\begin{aligned}
 f_t^\beta &= \sigma \left(W_{xf}^\beta \odot X_t^\beta + W_{hf}^\beta \odot H_{t-1}^\beta + W_{cf}^\beta \circ C_{t-1}^\beta + b_f^\beta \right) \\
 i_t^\beta &= \sigma \left(W_{xi}^\beta \odot X_t^\beta + W_{hi}^\beta \odot H_{t-1}^\beta + W_{ci}^\beta \circ C_{t-1}^\beta + b_i^\beta \right) \\
 C_t^\beta &= f_t^\beta \circ C_{t-1}^\beta + i_t^\beta \circ \tanh \left(W_{xc}^\beta \odot X_t^\beta + W_{hc}^\beta \odot H_{t-1}^\beta + b_c^\beta \right) \\
 o_t^\beta &= \sigma \left(W_{xo}^\beta \odot X_t^\beta + W_{ho}^\beta \odot H_{t-1}^\beta + W_{co}^\beta \circ C_t^\beta + b_o^\beta \right) \\
 H_t^\beta &= o_t^\beta \circ \tanh \left(C_t^\beta \right)
 \end{aligned} \tag{5}$$

A neural network model's architecture can vary according to the specific use case and application field. In this work, we designed the ConvLSTM NN architecture using the built-in Keras [53] ConvLSTM2D module [54, 55]. We developed two distinct model architectures for the ConvLSTM NN and trained both on the same dataset.

The first model, denoted as (1I/P-O/P), takes a single input frame and predicts the subsequent frame. In contrast, the second model, denoted as (3I/P-O/P), processes three successive frames as input and predicts the fourth frame. For a more detailed discussion of these two model architectures, refer to Section 2.4.

For both model architectures, the model hyperparameters are nearly identical except for the input layer.

The loss function is one of the most crucial parameters in the training of a neural network model, and its selection primarily depends on the type of prediction being made. In our case, dealing with a binary alloy system with two phases: c_A -rich and c_B -rich regions where c_A and c_B are the two components of the alloy. The phases can be represented by a single composition variable, c_A , since c_B can be expressed as $1 - c_A$. The range of this composition variable (c_A) varies from 0 to 1, where a value of 1 indicates a phase rich in c_A and a value of 0 indicates a phase poor in c_A . Inside the bulk region of the microstructure, after phase separation, the phase is either rich in composition c_A (i.e., $c_A = 1$) or c_B (i.e., $c_A = 0$). However, in the interface region, there is a compositional gradient from c_A to c_B (i.e., from $c_A = 1$ to $c_A = 0$) and vice versa. Leveraging this fact, we employed the Negative Log Loss function, also known as the Binary Cross Entropy Loss function, in our model. This function measures the

performance of the model by outputting probability values between 0 and 1, which aligns with the range of the compositional variable in our system. The expression for the Negative Log Loss function is given in Equation 6 [56]

$$\text{logloss} = -\frac{1}{N} \sum_{i=1}^N [y_{t,i} \cdot \log(y_{p,i}) + (1 - y_{t,i}) \cdot \log(1 - y_{p,i})] \quad (6)$$

where, $y_{t,i}$ and $y_{p,i}$ are the ground truth and prediction values, respectively.

Since the prediction task involves identifying phase separation in a binary alloy system with a single compositional variable varying from 0 to 1, the nature of the prediction aligns well with a binary classification problem, which is effectively handled by the log loss function. Additionally, the log loss function allows the model to output probabilities that represent the likelihood of the phase being rich or poor in c_A . This probabilistic interpretation is useful for capturing uncertainty and the interfacial region during phase separation.

2.4 Two Architectures Study

The final step before commencing neural network training is to categorize the training data into features and labels. Training features serve as the inputs to the neural network, while labels are used to verify the predicted results. Our training data consists of video sequences composed of image frames, and the goal is to predict each subsequent frame. Consequently, we treat each following frame as a label, while the preceding frame(s) serve as features. To achieve this, we created two different model architectures, denoted as A_1 : 1I/P-O/P and A_2 : 3I/P-O/P.

The distinction between these two architectures lies in the number of frames used as features during the training process. In A_1 , each individual frame (f_t) in a video sequence is input into the model to predict the adjacent frame (f_{t+1}). In contrast, A_2 uses three consecutive frames from a video sequence (f_t, f_{t+1}, f_{t+2}) as features to predict the fourth frame (f_{t+3}), as depicted in Figure 4. Both models were trained for 25 epochs with a batch size of 15 on a Dell Precision 7920 Tower equipped with 24 cores, 64 GB RAM, and an Intel Xeon(R) Gold 5220R CPU @ 2.20GHz × 48. Each model took slightly over an hour to complete a single iteration, with the entire training process spanning approximately 28 hours. It is worth noting that the A_2 architecture required more training time due to the larger input feature size. The choice of hyperparameters, including epoch size and batch number, was made to balance training time, processor performance, and training progress. The prediction performance of these two model architectures is discussed comprehensively in Section 3.

3 Results and Discussion

3.1 Quantification of Microstructure Prediction

Quantifying machine-learned predictions essentially involves comparing the prediction results with the corresponding Ground Truth (GT). In this work, we expanded

our approach from the default neural network metrics of accuracy and loss to include advanced image analysis techniques. One widely used method for image quality assessment is the Human Visual System (HVS). Although numerically quantifying visual perception is challenging, the HVS can provide a somewhat robust evaluation for distinguishing between visual stimuli [57]. However, since HVS is inherently based on individual perception, which varies according to biology and vision dynamics, this method can be subjective, inconvenient, and non-standard [58].

To overcome the limitations of HVS, we employed state-of-the-art, mathematically consistent error quantification methods, including the Microstructural Similarity Index (MSIM) [59], Blobs Detection, Shape Index [60], and Root Mean Squared Error (RMSE). These quantification tools were implemented using Python libraries such as *OpenCV* [61], *scikit-image* [62], *sewar* [63], and *pydom* [64]. Visualization of the quantitative analysis was performed with *Matplotlib* [65], which provided a means to observe and analyze deviations between the predicted results and the ground truth. This approach offers an objective method for quantifying physics-based neural network predictions, as opposed to the subjective nature of HVS. By employing these advanced metrics, our work establishes a new standard benchmark for future research. It represents one of the first state-of-the-art multi-model machine learning algorithms designed to predict spatio-temporal microstructural evolution as an image-tensor using physics-based phase-field simulations.

3.2 Best Predictive Model Selection

Two NN models, A₁ and A₂ as discussed in Section 2.4, were trained on the same Video Sequence dataset. These models differ in architecture: A₁ uses one frame as input to predict the subsequent frame, while A₂ uses three frames as input to predict the fourth frame. It is expected that these architectural differences will lead to variations in the predictions made by the two models. To numerically and scientifically quantify the predictions from these two NN models, we employed two image quality assessment methods and used a rigorous threshold measure to select the model for further analysis. The goal of this threshold measurement is to understand the differences in inferred microstructural evolution between the one-to-one prediction model (A₁) and the many-to-one prediction model (A₂). The Root Mean Squared Error (RMSE), a widely used error metric known for its simplicity in calculation and ease of interpretation, serves as the initial method for comparing these two NN models. The RMSE value between the predicted image frame and the ground truth (GT) image frame quantifies how far the pixel values in the predicted frame are from the corresponding pixel values in the GT frame. In Equation 7, $f_{(i,j)}$ and $f'_{(i,j)}$ represent the pixel values at spatial coordinates i and j for the GT frame and predicted frame, respectively, where both frames have the same pixel dimensions of $N \times M$.

$$RMSE = \sqrt{\frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [f_{(i,j)} - f'_{(i,j)}]^2} \quad (7)$$

For testing purpose a randomly selected 137 video samples from testing dataset was taken, which were used to evaluate both neural network (NN) model architectures. Root Mean Square Error (RMSE) was calculated for each video sequence, and the resulting RMSE values were plotted against the video sequence numbers, as shown in Figure 3a for model architecture A₁ and Figure 3b for model architecture A₂. The average RMSE values across all 137 video sequences in Prediction Generation Zero (GEN-0) and Prediction Generation One (GEN-1) were 0.1 and 0.34, respectively, for architecture A₁, while for architecture A₂, they were 0.06 and 0.09. It is important to note that in this context, GEN-0 prediction refers to predictions made by the NN model using input solely from the phase field simulation, whereas GEN-1 predictions incorporate inputs derived from GEN-0 predictions as well. For a more detailed discussion on generational prediction, refer to Section 3.3.

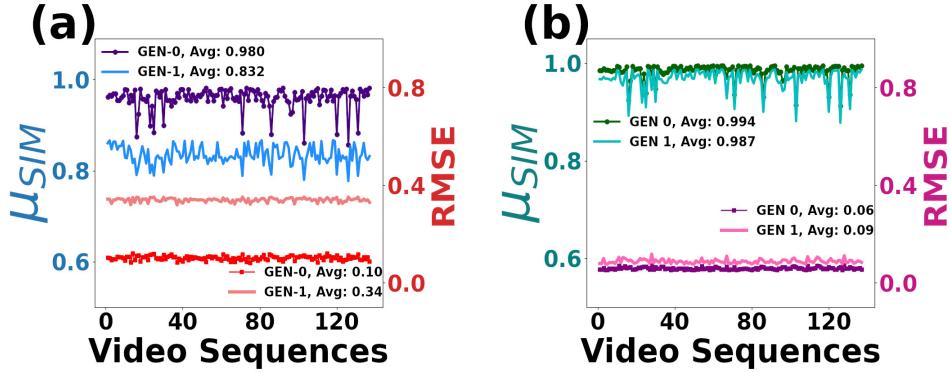


Fig. 3 Prediction result quantification for selection of best model among two purposed model architectures. a) Microstructural similarity index along with root mean squared error comparison of Model architecture A₁ for two successive generations (GEN-0, GEN-1) of predictions, The Left Y-axis represents μ_{SIM} index with a blue shade color of graph and Right Y-axis represents RMSE value with red shade of color. The average μ_{SIM} and RMSE value for GEN-0 prediction for A₁ is 0.98 and 0.1 respectively, as we approach to GEN-1, MSIM plummets to 0.812 where as RMSE rockets up to 0.34. b) Result of μ_{SIM} and RMSE same successive generation for Model architecture A₂. It is observed in prediction result of A₂ that the drop and rise of μ_{SIM} and RMSE is not that significant in two generations as compared to A₁ indicating the robustness of A₂ architecture. For both models the predictions were performed for exact same 137 validation video-sequences as represented by X-axis in both plots.

The second metric used as a quantifier is the Microstructural Similarity Index (μ_{SIM}). μ_{SIM} assesses image quality by comparing similarities in terms of luminance, contrast, and pixel structure between two images. The mathematical expression to calculate μ_{SIM} is presented in Equation 8, originally formulated by Wang et al. [58] as the Structural Similarity Index (SSIM) to quantify similarities between two image frames. In this study, we adapted Wang et al.'s similarity index for the specific purpose of microstructural quantification, thus naming it the Microstructural Similarity Index (μ_{SIM}) [66]. In the equation, f_t represents the ground truth microstructural image frame, and f'_t denotes the predicted microstructure frame at time 't'.

$$\mu_{SIM}(f_t, f'_t) = \frac{(2\mu_{f_t}\mu_{f'_t} + \alpha_{f_t})(2\sigma_{f_t f'_t} + \alpha_{f'_t})}{(\mu_{f_t}^2 + \mu_{f'_t}^2 + \alpha_{f_t})(\mu_{f_t}^2 + \mu_{f'_t}^2 + \alpha_{f'_t})} \quad (8)$$

The symbols μ_{f_t} , $\mu_{f'_t}$, $\sigma_{f_t}^2$, and $\sigma_{f'_t}^2$ represent the pixel sample mean intensities and variances of frames f_t and f'_t , respectively, while $\sigma_{f_t f'_t}$ denotes the covariance between the two image frames. The terms α_{f_t} and $\alpha_{f'_t}$ are constants introduced to prevent division by zero errors. One crucial condition for obtaining an accurate MSIM index is that both frames, f_t and f'_t , must have equal pixel sizes of $N \times M$.

Similar to RMSE, we also calculated the MSIM for the predictions of both models on the 137 validation video sequences. The MSIM values for each video sequence are plotted in Figures 3a and 3b, categorized into two generations: GEN-0 and GEN-1, for the two NN model architectures A₁ and A₂. It is observed that the MSIM index decreases while the RMSE value increases with each generational step. Between the two architectures, A₁ and A₂, the latter, which takes three frames as input, appears to perform better. The drop in MSIM and the increase in RMSE values in model architecture A₁ upon reaching GEN-1 are more significant than those in architecture A₂, suggesting that the second architecture is more robust and reliable for generational prediction.

Based on the analysis of the results from the two metrics, RMSE and MSIM, we decided to select model architecture A₂ as our best model and proceed further with this model for the further investigation.

3.3 Multi-Generational Prediction with μ_{SIM}

One of the key aspects of any machine learning or predictive model is to assess and quantify how accurately its predictions hold over successive time steps. In this work, which deals with inferring microstructure evolution, we have assessed generational prediction and quantified the evolution of microstructure in successive generations of prediction using various image correlation metrics. These metrics include the structural similarity index [58, 66, 67], the ratio of blob detection [68], and the root mean squared error (RMSE) between predicted and ground truth images.

As illustrated in Figure 4a, the first row of the figure shows the model taking three frames (f_1 , f_2 , f_3) of image data generated from the simulation results of the Phase Field Model produced by the MOOSE Framework, which is driven by the physics of phase decomposition. The prediction result (f'_4) from the model is considered as the generation zero (GEN-0) prediction with no cross-breeding, where every input (parent) to the model is a physics-backed simulation result. For the next generation (GEN-1) prediction, we use two frames (f_2 , f_3) from the simulation results and one frame from the machine-predicted zeroth generation result (f'_4) as the parent image frames (input image frames) to the model. By using the prediction of the previous generation as the parent for the next generation, we cross-breed between machine learning and physics-based results, transitioning from purely physics-based inference to physics-informed machine learning methods. In this way, the first-generation prediction result (f'_5) inherits the characteristics of its parent frames (f_2 , f_3 , and f'_4). Similarly, for the second-generation result (f'_6), we use one frame from the simulation result (f_3), one from GEN-0 (f'_4), and one from GEN-1 (f'_5) as input parents to the model,

which predicts the result as GEN-2 inference (f'_6). At each generation, we progress along a path of computational reducibility by transitioning more toward machine-predicted microstructural evolution. On the third generation, GEN-3, we complete the transition and fully intake three image frames as parent frames from each prediction result of each generation: GEN-0: f'_4 , GEN-1: f'_5 , and GEN-2: f'_6 . The third and final generation results in frame f'_7 , which is the culmination of predictions built upon previous predictions.

It can be observed that the structural similarity index drops by a certain fraction in every successive generation from GEN-0 to GEN-3. This is because each subsequent predicted generation inherits certain microstructural characteristics from its parent generation, including the differences between ground truth and the model's prediction, thus differing slightly more from the ground truth with each generation.

The approach of using three frames as features to predict the next frame allows us to address kinetics. As the model is trained on a large number of video sample sequences comprising different simulation parameters, as discussed in Section 2.1, our model architecture A_2 is capable of predicting successive next frames by intelligently learning the time step difference Δt between the three input image frames. To address the variable kinetics of the reaction, we strategically curated training video sequences with different frame steps δ ranging from 1 to 15. In every video sequence, δ plays a crucial role in determining each subsequent frame of the image to incorporate. By varying δ within the range of 1 to 15, we were able to virtually speed up the phase field simulation's kinetic parameters.

We further investigated the accuracy and robustness of the selected NN model architecture A_2 through an in-depth analysis of the Microstructural Similarity Index (μ_{SIM}). Individual assessments of the μ_{SIM} index for all four generational predictions were conducted using the validation video samples, and a comparative radial line graph was plotted, as shown in Figure 4b.

It is observed that the similarity index is highest for the zeroth generation prediction (GEN-0), with an average μ_{SIM} value of 0.9942 across 137 video samples. A visible pattern emerges in the graph, indicating that the μ_{SIM} index drops slightly from 0.9942 in GEN-0 to 0.9874 in GEN-1, further to 0.9806 in GEN-2, and finally to 0.9672 in GEN-3. We have concluded that this gradual decrease in the μ_{SIM} index value in every successive generation is due to the inheritance of microstructural prediction features from one generation to another. These inherited features can contribute either constructively or detrimentally, depending on the specific prediction being made. As our NN model utilizes prior generational prediction results as input for subsequent generation predictions, it establishes a parent-child relational dynamic with feature inheritance between each generational prediction.

Since, after GEN-3, the model's predictions become independent of the input frames generated from the phase field simulations and rely solely on the results of previous predictions, this phase is referred to as the 'prediction-based prediction' zone. Our analysis shows that the model achieves a microstructural similarity index of 90% or higher for up to the 10th generation of prediction as illustrated in Figure B.1. This corresponds to an estimated computational time of 273.25 seconds. Beyond the 10th

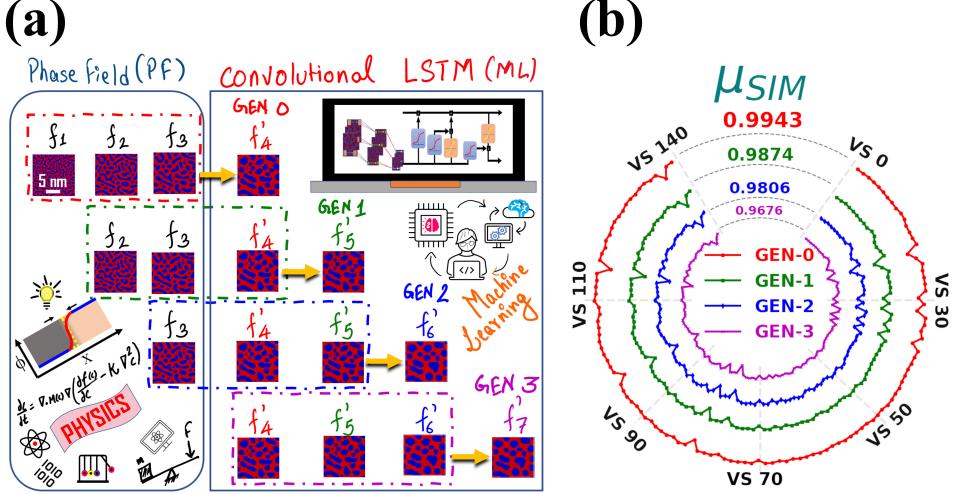


Fig. 4 (a) A flow diagram visualizing multi-generational microstructural prediction for architecture A₂. The first red colored dotted box-row represents generation zero (GEN 0) prediction where 3 input frames (f_1, f_2, f_3) are supplied to the trained model, to predict fourth frame f'_4 , at this generation all input frames (15nm × 15nm each) are generated from phase field simulation. On reaching generation one (GEN 1), i.e. second row green dotted box, two frames (f_2, f_3) are taken from phase field result in combination with frame (f'_4) of GEN 0, to predict fifth frame (f'_5). Moving on to generation two (GEN 2) i.e. blue colored dotted box on third row, we take one frame (f_3) given by simulation alongside two predicted frames (f'_4, f'_5) from GEN 0 and GEN 1 as input, this results in prediction of sixth frame (f'_6). At every generation, input to the trained model is shifting from physics informed phase field generated microstructural image-frames towards previous image-frames predicted by the model itself on previous generation. And lastly, at generation three (GEN 3) the magenta colored dotted box at fourth row, we completely shift model's input from phase field simulated frames to generational predicted frames (f'_4, f'_5, f'_6). The outcome of GEN 3 i.e. seventh frame (f'_7) is purely a result of nested machine learning framework, hence making it a prediction based on predictions. At this step we are fully in the territory of Convolutional LSTM region (right rectangular box of Machine Learning) leaving the physics backed region (left rectangular box of Physics) of Phase Field. (b) Radial line graph of Microstructural Similarity Index (MSIM) of microstructural evolution in four successive multi-generational prediction among validation video sequences. It is observed that MSIM value drops on every next generation starting from GEN-0 with average value of 0.9942 to GEN-3 with 0.9672. This drop in structural similarity index can be considered as a consequence of children generation inheriting prediction inaccuracies from the parent generation.

generation, however, the model's accuracy, measured by the microstructural similarity index, falls below the 90% threshold. Therefore, we do not recommend using the results beyond this point for actual decision-making processes in failure analysis.

3.4 Microstructural Blobs Detection

We further applied a priori algorithm to detect microstructural blobs present on the alloy surface. Blobs refer to patterns or regions in a microstructure surface with different compositional properties compared to their surroundings. Numerically quantifying the blobs present on the surface can serve as a valuable tool for studying imperfections

$$(LoG) : \nabla^2 G(M, N) = \frac{1}{\pi S_\sigma^4} \left(1 - \frac{M^2 + N^2}{2S_\sigma^2}\right) \exp\left(-\frac{M^2 + N^2}{2S_\sigma^2}\right) \quad (9)$$

$$DoG = LoG(M, N, S_{\sigma 1}) - LoG(M, N, S_{\sigma 2})$$

on the alloy surface. This approach can be used to detect cracks in the bulk phase, precipitates in the matrix phase, or voids in a grain boundary.

The detection of differences in patterns or blobs occurs when there is a sudden change in pixel intensity (i.e., crossing an edge) while convolving a kernel within an image. For blob detection, we applied the Laplacian of Gaussian (LoG) Kernel to both the ground truth (GT) image frame and the predicted frame. LoG has a well-established history of being used to detect patterns and blobs in highly sensitive scientific fields such as medical imaging [69–71]. The expression for LoG calculation is represented by Equation 9 [72, 73], where M and N are positional image coordinates with certain pixel intensities and a specified spatial constant S_σ . Another pattern-detecting algorithm is the Difference of Gaussian (DoG), also represented in Equation 9. DoG is primarily an approximation of the scale-normalized LoG itself [74], where $S_{\sigma 1}$ and $S_{\sigma 2}$ represent the standard deviations of two Gaussian functions selected by the user to capture global and local microstructural details in an image. Figure 5a displays the application of the LoG and DoG algorithms to a randomly selected frame from one video sample.

Moreover, our study primarily focused on using the Laplacian of Gaussian Kernel to analyze the rate of blobs detected in different frames of predicted images compared to the ground truth image frame.

We began by implementing the Laplacian of Gaussian (LoG) Kernel [75] on image-frames from ten randomly selected validation video samples using the *Scikit-Image* library [62]. Similarly, for the same selected video frames, we conducted predictions for four generations of microstructural evolution and subsequently applied the LoG Kernel to them. We chose the ratio of blobs detected in the Phase Field-generated image-frames compared to that in the NN-predicted image-frames as the metric to assess the robustness of our model. An ideal prediction model would yield predicted image frames with an equal number of blobs as in the ground truth (GT) image-frames, resulting in a ratio of blob detection equal to 1. This information allows us to further analyze our model's performance in generational microstructure prediction.

The lower section of Figure 5 displays the ratio of blobs detected in ten video samples, each containing 17 frames (a total of 170 frames), for four generations of microstructure prediction. In Figure 5b, it can be observed that GEN-0 exhibits the least deviation (± 0.032) from the mean (0.98), while the standard deviation continues to increase with each successive generation. Figure 5c corresponds to GEN-1, Figure 5d to GEN-2, and, finally, the most significant deviation in the blobs detection ratio can be observed in the final generation of the study, GEN-3, as shown in Figure 5e. The increasing deviation in blobs detection over the four generations suggests that certain generational traits are being carried over in each generation, and a threshold

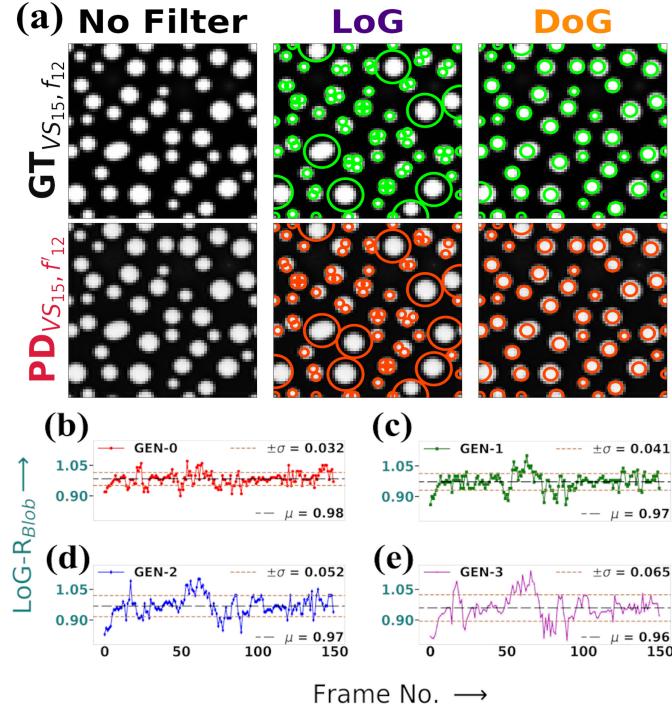


Fig. 5 Graphical comparison between microstructural blobs present in neural network predicted, and phase field simulation generated image-frame alongside the quantitative study of multi-generational blobs detection ratio in ground truth to predicted microstructure. (a) the first row consists of ground truth frame 12 (f_{12}) of video sequence 1 (VS₁) with no filter, Laplacian of Gaussian (LoG) filter and Difference of Gaussian (DoG) filter applied on each individual columns, while second row represents same filters applied along each column for predicted frame 12 (f'^{12}) of video sequence 1 (VS₁). (b) Ratio of blobs detected in image frames of GEN-0 predictions to GT image frames. GEN-0 shows the least deviation (standard deviation of 0.032) in detecting blobs in comparison to that of blobs present in simulation result where as the deviation in detection is found to increase on every successive generational prediction (c) GEN-1 with deviation of 0.041 (d) GEN-2 with deviation of 0.052 and on reaching GEN-3 (e) the final generation considered for this study it is observed to have the maximum deviation of 0.065 from the mean blobs detected. For all four generations, image frames are taken from 10 randomly sampled validation video sequences.

deviation could be established, above which we may consider discontinuing generational prediction. In this case, even though the fourth generation (GEN-3) has the highest deviation (± 0.065) compared to previous generations, the value itself is not particularly significant, making generational analysis an interesting consideration.

3.5 Histogram of Gray Pixels

A histogram is a visualization tool used to represent the frequency of occurrence in a particular subject of study. It can serve as a quantification measure by comparing the histogram graphs between different subjects. A histogram of gray pixels refers to the distribution of pixel intensity values in a grayscale image frame. By definition, grayscale image pixel values range between various shades of gray, with a maximum

value of 255 for white and a minimum of 0 for black. Therefore, a histogram of a grayscale image provides information about the number of occurrences of specific pixel values in that image. We plotted the histogram chart to compare the generational predictions of microstructural evolution with the ground truth image frame, as shown in Figure 6a.

In the histogram graph, we can observe that the microstructural evolution data exhibits a bimodal distribution. There are two separate peaks skewed towards opposite ends of the pixel values (0 and 255), corresponding to two different phases. It can also be inferred that among the four prediction generations, GEN-0 (red line) provides the most accurate representation of the ground truth, with the highest frequency of pixel values at the opposite ends. This is followed by GEN-1 (green line), GEN-2 (blue line), and finally GEN-3 (magenta line). This observation aligns with our previous quantification measures of μ_{SIM} and blob detection ratio.

It is worth noting that there are relatively few cases of extreme pixel values (0 or 255, black or white) compared to the frequency of pixel values near black (between 25-50) and near white (between 200-240). This suggests that the microstructure is in a transitional phase of decomposition, where it has yet to fully separate into two opposite phases. Histogram analysis can also assist in determining whether the decomposition reaction is complete or ongoing by observing the peak frequency of pixel intensity values.

Figures 6b and 6c present a sample case study for pixel intensity comparison at specific pixel coordinates between two image frames, one from the ground truth and the other from GEN-0 prediction. It is evident that GEN-0 prediction maintains a well-coordinated result of pixel intensities when compared between two different positions with opposite intensities.

3.6 Shape Index (σ_t)

In the context of surface shape classification, a combination of Mean Curvature (H) and Gaussian Curvature (K) is used to categorize a point on the image surface into distinct shapes such as Elliptical, Hyperbolic, Parabolic, or Planar. This classification is based on the principal curvatures k_1 and k_2 , where $K = k_1 k_2$ and $H = \frac{k_1 + k_2}{2}$ [76]. Since our study focuses on analyzing the interfacial region where the phase transitions from one to another, we require an index that can intuitively describe the region where phase transformation occurs, specifically the diffused interface region. This index should account for the interfaces and boundaries of local curvature within the microstructure. For such an analysis, a single value, dimensionless number that fulfills the quantification criteria is the Shape Index (σ_t) as proposed by Koenderink et al., [60]. They decoupled the shape from the magnitude of curvedness, making the surface curvature invariant to changes in scale. The Shape Index provides a continuous measure of the shape of a surface at a point, ranging from -1 to +1. The expression for calculating the Shape Index is given in Equation 10 [77].

$$\sigma_t = \frac{2}{\pi} \arctan \frac{k_2 + k_1}{k_2 - k_1}; \quad (k_1 \geq k_2) \quad (10)$$

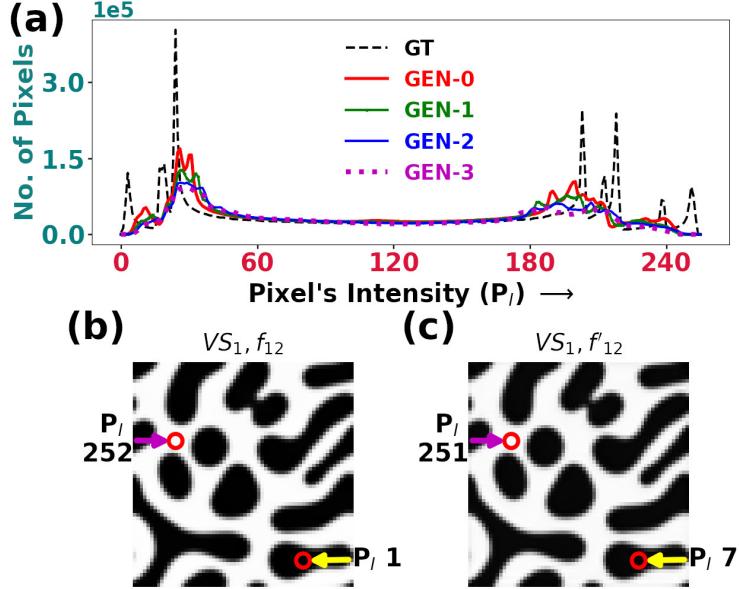


Fig. 6 a) Histogram plot between Ground Truth (black line) image frame of all validation video-sequences and four generational predictions GEN-0: red, GEN-1: green, GEN-2: blue and GEN3: magenta. Histogram curve flattens out as generation progresses in time indicating predictions will be of lower intensity and diffused image-frame. b) A ground truth image frame from validation video-sample no 1, frame no 12. Image pixel intensities at two random coordinate points with opposite shades of gray are found to be $P_i: 252$ (near white region) and $P_i: 1$ (near black region). c) GEN-0 predicted image-frame for exactly same video-sequence and frame number with pixel intensities of $P_i: 251$ (near white region) and $P_i: 7$ (near black region) calculated at the exact same coordinate point to that of ground truth image frame.

where,

$-k_1$ and k_2 are the maximum and minimum principal curvatures, calculated as the eigenvalues of the Hessian matrix.

In this work, we have employed the Shape Index to identify and locate curvatures in the microstructure by analyzing distinctive interfacial characteristics. A positive Shape Index value corresponds to regions with a convex shape, whereas a negative Shape Index indicates a concave shape. A Shape Index value of zero ($\sigma_\mu = 0$) signifies the presence of an edge or a transition between different shapes. Since an interface represents a change in metastable phase curvature or a transition from one phase to another, a Shape Index value of zero serves as a reference point for correlating the interface between the predicted microstructure and the ground truth frame. The calculation of the Hessian matrix and the subsequent Shape Index value is performed using the open-source Python library Scikit-Image. [62].

Figure 7 illustrates the curvature features and interface regions detected using the shape index target $\sigma_t = 0$. Interface detection is performed for both the ground truth and predicted frames of the same microstructural frame. The magenta dots surrounding the microstructure's interface denote points that deviate by no more than 0.3 from the targeted value of 0.

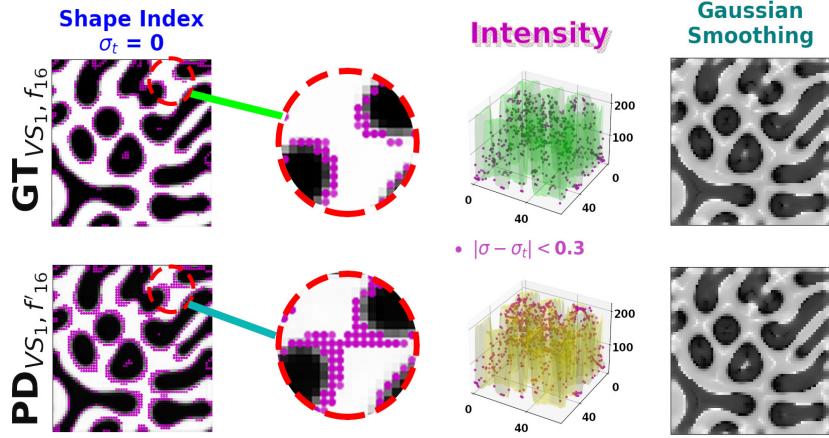


Fig. 7 Shape index could be potentially utilized as a tool for interface informatics. Microstructural interface detected in ground truth and predicted image for a randomly selected microstructural frame of a video sequence using the shape index with target value of $\sigma_t = 0$. The difference detected in the magnified part of GT and PD suggest the predicted microstructural image frame doesn't have a clearly defined interface compared to the ground truth and therefore there might be phase growth corresponding to darker region in future time step. The magenta dots in the shape index diagram suggests that these regions are the ones which deviate from the targeted shape index value of 0 no more than 0.3 suggesting that the interfacial region is of smaller width than shown by the magenta dots. The value of 0.3 as cutoff deviation is chosen for visualization purpose as the dots becomes finer and difficult to visualize for deviation near to zero value. The 3D intensity graph represents the microstructural pixel intensity ranging from 0 to 255 in a three dimensional framework with interface denoted by magenta dots in it. Gaussian smoothing performed for detection of interface via shape index to attenuate the image noise present in microstructure frame, this smoothing technique also helps in finding the sharp change in interface by showing these regions in contrasting colors such as white in dark background.

The magnified section inside the dotted circle provides a closer look at the difference between the interfaces detected in the ground truth and predicted frames. It is evident that, in the predicted frame, the interface appears larger, suggesting potential phase growth in that region in future time steps. The 3D graph in the center displays the intensity of microstructural pixels in a three-dimensional framework, with magenta dots marking the interface regions within this 3D space. On the rightmost column, the microstructural images for both the ground truth and predicted frames are shown after passing through a Gaussian smoothing kernel. Gaussian smoothing is applied to reduce and attenuate noise present in the microstructural image data before determining the shape index. The Gaussian smoothing of the microstructural image frame, used for detecting interfaces with a shape index value of zero, visually provides information related to interfacial changes. In the figure, a bright spot within the dark phase at the center of the microstructure is highlighted by the magenta dots in the shape index diagram.

4 Interface in microstructures of Pd-Rh and Cu-Mn alloys

We employed the trained model to rapidly forecast microstructural changes in two distinct engineering alloy systems: Pd-Rh and Cu-Mn. Phase field simulations were conducted for both alloy systems using the MOOSE Framework. The values of the driving force ($G_{min,Cu-Mn}$ and $G_{min,Pd-Rh}$) were determined from thermodynamic data based on CALPHAD at temperatures of 586.15 K and 1196.0 K, respectively. These temperatures were chosen with reference to phase stability decomposition studies conducted by Saha et al. [78] and Garcia et al. [79].

The landscape of the Gibbs free energy curve for both the Pd-Rh and Cu-Mn alloy systems is dictated by the coefficients of Equation 3, as shown in Figure 8a for the Cu-Mn alloy system and Figure 8b for the Pd-Rh system. The values of the six coefficients corresponding to Equation 3 for the Cu-Mn and Pd-Rh systems are presented in Table 2.

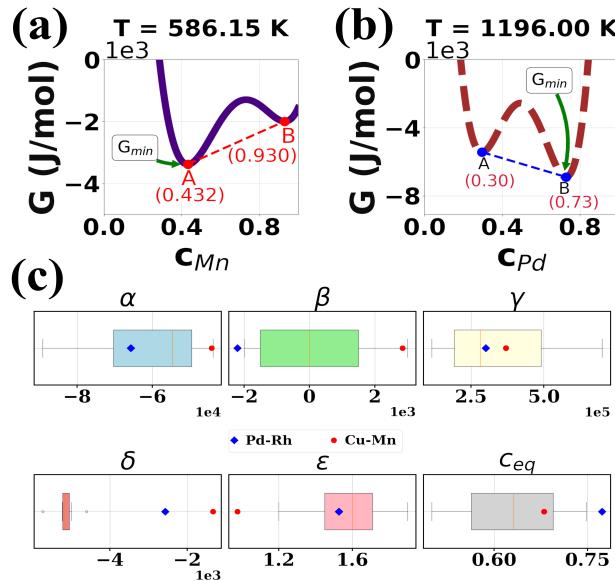


Fig. 8 a) The landscape of double well curve representing Gibbs free energy for Cu-Mn binary alloy system at 586.15 K with x axis denoting composition of Mn. The two points A ($c_{Mn} = 0.432$), and B ($c_{Mn} = 0.930$) denotes two minima points in free energy curve, where global minima of free energy lies at the composition $c_{Mn} = 0.432$ with a value of $G_{min} = -3.378e3$ J/mol. Similarly, b) represents the free energy curve for Pd-Rh alloy system at 1163 K with two points A ($c_{Pd} = 0.30$) and B ($c_{Pd} = 0.73$) representing two wells of double well curve with a global minima of free energy $G_{min} = -6.893e3$ J/mol at composition $c_{Pd} = 0.73$. c) Box plot visualizing the variation in values of six different coefficients used in Equation 3 to create the training data for the model along with the values of same coefficients for Pd-Rh system (represented in blue diamond shape) and Cu-Mn system (represented in red dot).

Table 2 The values of coefficients (features) for Cu-Mn and Pd-Rh binary alloy system that subsequently create the phase field generated input in the prediction model. It is to be noted that the magnitudes of these variables were determined from the CALPHAD information. For binary Cu-Mn alloy α , β , γ and c_{eq} are well within the features range ascribed in training data whereas, δ and ϵ clearly outside the training data range. Similarly, for Pd-Rh system β , δ and c_{eq} fall outside the range of training values whereas α , γ and ϵ are within the range.

Alloy	α (J/mol)	β (J/mol)	γ (J/mol)	δ (J/mol)	ϵ	c_{eq}	G_{min} (J/mol)
Cu-Mn	-4.4232e4	2.8438e3	3.6976e5	-1.339e3	0.9743	0.68	-3.378e3
Pd-Rh	-6.5701e4	-2.2055e3	3.0077e5	-2.568e3	1.527	0.7733	-6.893e3

By leveraging the four microstructural analysis tools discussed in Section 3, we can quantify the errors at the interface level of the predicted microstructure in these two engineering alloy systems. The Microstructural Similarity Index (μ_{SIM}) measures differences in contrast, luminance, and structure at an interfacial resolution to assess the predicted microstructure in comparison to the ground truth image frame.

Blobs detection is utilized to identify regions in the microstructure where there are notable changes in background contrast, indicating an edge or interface of a microstructural phase. Additionally, the Shape Index is employed to detect the entire region of interfaces present in the microstructure by targeting a shape index value of σ_t to 0.

Similarly, the histogram of pixel intensity graph can provide insights into prediction accuracy, interfacial width, and the clear separation of phases in the microstructure.

4.1 How similar are the learned microstructures?

To quantify our model’s predictions in real engineering alloy systems, we initially analyzed four consecutive generations of microstructural evolution predictions using the μ_{SIM} index. The calculation of the Microstructural Similarity Index (μ_{SIM}) was performed for a randomly sampled set of 100 video sequences generated using our proprietary algorithm, as discussed in Section 2.1, for both alloy systems.

Figure 9a displays a linear graph plotting the μ_{SIM} values for four generations of predictions in the Cu-Mn alloy system. The graph illustrates that for the first prediction generation (GEN-0), the μ_{SIM} value is the highest at 0.917, but it gradually decreases to 0.871, 0.829, and 0.792 in the subsequent three generations. In contrast, for the Pd-Rh system, as shown in Figure 9b, the μ_{SIM} value for the first generation is higher at 0.976, and the microstructural similarity index does not decrease as rapidly for the next three generations, with values of 0.958, 0.941, and 0.916, respectively.

In the preliminary stage of quantifying microstructural evolution predictions in real engineering alloys, our machine learning model appears to predict the Pd-Rh alloy system with greater accuracy than the Cu-Mn system.

4.2 Locating the edges

Upon further investigation of blobs detected in the microstructural surface for predicted frames and ground truth image frames in a randomly sampled set of 140 microstructural images, using the technique discussed in Section 3.4, some interesting findings emerged. For the Cu-Mn alloy system, the number of blobs detected in the

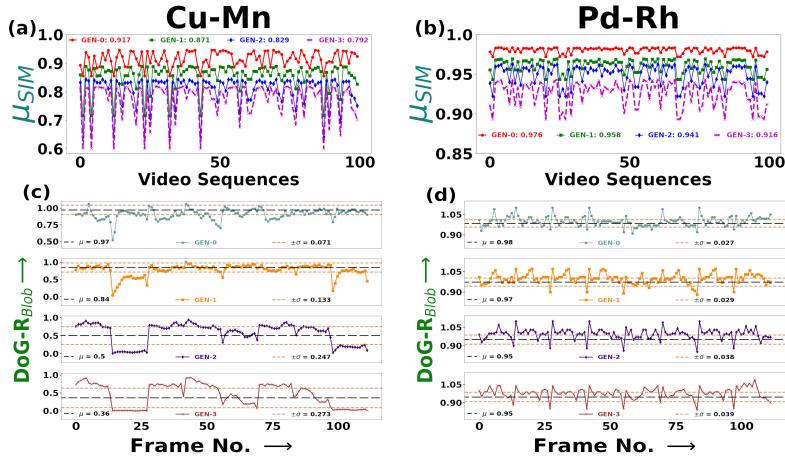


Fig. 9 a) Microstructural similarity index (μ_{SIM}) plotted for randomly samped 100 video sequences in four generations of predicted microstructure for Cu-Mn binary alloy system. The consecutive μ_{SIM} values for four generations are 0.917, 0.871, 0.829 and 0.792 represented by red, green, blue and magenta lines respectively. b) Linear plot of μ_{SIM} in generational prediction result of Pd-Rh alloy system for randomly selected 100 video sequences, the four generation of prediction are represented by red, green, blue and magenta colored lines with μ_{SIM} values of 0.976, 0.958, 0.941 and 0.916 respectively. c) The ratio of blobs detected in predicted frame of microstructure to that of ground truth frame for four consecutive generation of predictions made by the model for Cu-Mn alloy system using difference of gaussian kernel. The teal colored line represents first generation of prediction and has the least amount of blobs detection deviation where as the yellow, purple and brown colored graph corresponds to second, third and fourth prediction generation with increasing value of deviation for number of blobs detected. d) Ratio of blobs detected by the difference of gaussian kernel for Pd-Rh alloy system in four consecutive microstructure prediction generation. Although deviation in blobs detected from ground truth frame to that of predicted frame increases from 0.027 to 0.039 from first generation to fourth generation, the change in deviation is quite less significant compared to that in Cu-Mn alloy system.

predicted microstructure image matches that of the simulation-generated frame in the first generation, with a mean ratio exceeding 0.97 and a deviation of 0.071. However, the mean ratio drops to 0.84 in the second generation and plummets further to 0.5 and 0.36 in the third and fourth generations, as illustrated in Figure 9c. This suggests that the model provides fairly accurate predictions up to the second prediction generation for the Cu-Mn alloy system.

In contrast, for the Pd-Rh alloy system, the results are significantly different. The ratio of blobs detected in multiple prediction generations for Pd-Rh remains consistently high, with a mean value for blobs detected in the microstructural surface in all four consecutive generations exceeding 0.95 and a maximum deviation of 0.039 observed at the end of the fourth generation. This further supports the finding that the model predicts remarkably well for the Pd-Rh alloy system compared to the Cu-Mn system.

Several factors may contribute to our model's superior performance in the Pd-Rh alloy system compared to the Cu-Mn system. One such factor is the value of the driving force (G_{min}), which is significantly higher for Pd-Rh (-6893.89 J/mol)

compared to Cu-Mn (3378.08 J/mol). This suggests that the model performs better for alloy systems with G_{min} values in the range of 10e4.

Additionally, the model's performance can be correlated with the landscape of the Gibbs free energy curve for both alloy systems, as shown in Figure 9a and b. For an alloy system where phase separation occurs near the equal composition region of the constituent elements, the model appears to predict microstructural evolution more accurately.

4.3 Spatially positioning the region of microstructural transition: Ingenuity of Shape Index

As discussed in Section 3.6, the shape index can be employed to detect microstructural interfaces by targeting a shape index value equal to zero. The shape index also assists in quantifying differences in microstructural patterns and changes occurring at phase boundaries. For both the Pd-Rh alloy and Cu-Mn system, we randomly selected a single frame of microstructure from the set of video sequences and compared the results of interface detection in predicted and ground truth frames.

Figure 10a illustrates the microstructural interface detected for the Pd-Rh system. In the zoomed area, it is evident that there is a certain region in the predicted frame detected as the microstructure interface, indicating the potential development and growth of a phase boundary in future time steps.

Similarly, in Figure 10b, the interface detected for the Cu-Mn alloy system is shown. The magnified portion highlights that the predicted frame still contains a small dark region acting as a phase interface, while this region is absent in the ground truth. Using the shape index, we can not only compare and quantify the phase interface between predicted and simulation-generated microstructural frames but also detect regions where interfaces may develop in future time steps.

4.4 Detecting errors of prediction

The third and final quantification measure used to assess the model's performance in two engineering alloy systems (Pd-Rh and Cu-Mn) is the histogram of pixel intensity. Figure 10c presents a twin-plot of the histogram of pixel intensity in randomly selected 100 video frames for both alloy systems.

It is noticeable that for both the Cu-Mn system and Pd-Rh system, the histogram curve for the ground truth microstructural frame peaks at the two extremes of intensity values, near 0 and near 255, represented by the green and black solid lines, respectively. However, when examining the histogram of the first generational predicted frame for the Cu-Mn system, it becomes evident that the curve deviates away from the extremes and toward the center for GEN-0 and GEN-3, represented by the teal colored solid line and yellow dashed line. This deviation may be attributed to the poor prediction of the bulk microstructure phase along with wider interfacial region predictions.

Since the microstructural interface represents the region where there is a change in pixel intensity, transitioning from white to gray or gray to white microstructural patterns, a histogram curve with a high number of pixels in the middle pixel intensity region suggests the presence of a wider interface with less distinct bulk phases. In

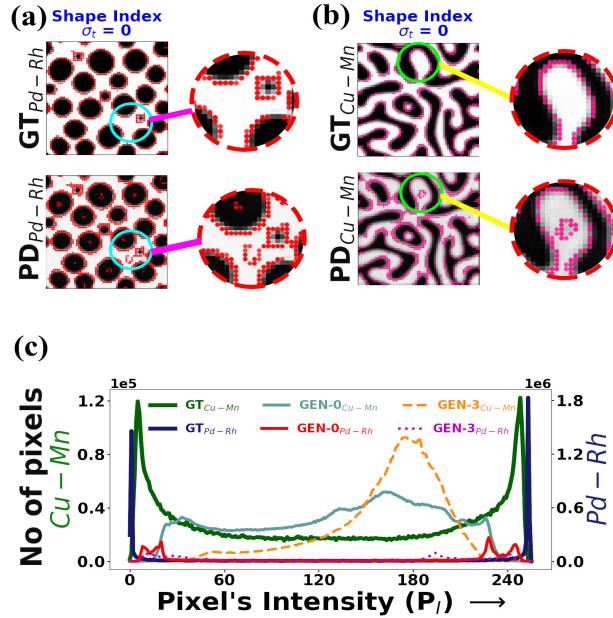


Fig. 10 a) Detection and comparison of microstructural phase-interface in Pd-Rh engineering alloy system between ground truth image and model predicted image of a randomly selected microstructural frame by the application of shape index analysis. The magnified part denoting the difference in interface detected in ground truth and predicted frame where, a small extra interfacial region is highlighted in predicted frame by the shape index. Shape index value ranges from -1 to 1 with two extreme denoting either concave or convex microstructural pattern, whereas shape index of value 0 denotes the interface or edge in the pattern, hence setting the target of shape index $\sigma_t = 0$ we are able to detect the microstructural interface in the image frame. b) Similar analysis of interfacial region detected for Cu-Mn alloy system with a highlight showing an interface in predicted frame which is absent in ground truth meaning the predicted result is lagging behind. c) A twin plot representing the histogram of pixel intensity for predicted and ground truth microstructural frames in two real alloy system. The left y-axis represents Cu-Mn alloy system with a green colored solid line for ground truth frame's pixel histogram which lies mostly at the extreme values of intensity 0 and 255 (0 for black region, 255 for white). The first and fourth generation prediction result is represented by teal solid line and yellow dashed line which deviates from the extreme intensity value meaning the prediction result has wider interface with not so distinct phase separation. Meanwhile the right y-axis represents Pd-Rh alloy system with purple solid line for ground truth microstructural frames that rises at the two extremities of 0 and 255. The major difference between Cu-Mn and Pd-Rh system lies in the generational prediction result as the first and fourth generation denoted by red solid line and magenta dotted line also mimics the ground truth result by rising near the extremities meaning the interface width is shorter and there is clear phase separation in this system.

contrast, for the Pd-Rh alloy system, the first (red solid line) and fourth (magenta dotted line) generational predictions closely align with the ground truth microstructural image, as they do not deviate towards the middle range of pixel intensity and remain near the extreme values. This further validates the model's superior performance for the Pd-Rh alloy system compared to the Cu-Mn system.

5 Conclusion

In conclusion, this research work establishes a benchmark model for predicting multi-generational microstructural evolution using state-of-the-art Conv-LSTM neural networks. By capturing spatio-temporal dependencies with remarkable accuracy, the proposed neural network architecture demonstrates its robustness and computational efficiency in predicting microstructural changes in binary alloy systems undergoing phase decomposition. Transitioning from finite element analysis to data-driven machine models for inferring microstructural behavior, this study successfully achieves the objective of computational reducibility, yielding high microstructural similarity indices. It offers a comprehensive insight into the prediction capabilities of the trained model through various quantification techniques, including μ_{SIM} , blob detection ratio, histogram-based pixel intensity comparisons, and shape index differences.

The significance of this research work lies in its contribution to the field of spatio-temporal behavior prediction of microstructures, where accurate inferences are of utmost importance for engineering applications. Notably, the model not only performs well on the testing dataset but also accurately predicts the microstructural evolution and interfacial changes in real engineering alloys, specifically in the Pd-Rh and Cu-Mn systems. A comparison between these two binary alloy systems reveals a superior performance of the model in predicting the Pd-Rh alloy system, with driving force differences playing a major role in dictating model accuracy.

These findings demonstrate the potential applications of these tools in high-risk scenarios to enable early action in preventing hazards and accidents by predicting material deterioration at the microstructural level. The combination of quantitative evaluation metrics with Conv-LSTM model predictions also paves the way for future research to uncover previously undiscovered material behaviors, facilitating the design and optimization of advanced materials. This work opens up new possibilities in the integration of machine learning models with image analysis techniques, providing insights into dynamic and intricate microstructural evolution behavior. Furthermore, it advances the field of materials science and engineering by enabling accurate and efficient predictions of multiple generations of microstructural patterns.

Subsequent progress and further enhancements of the current predictive model are possible and can be achieved by implementing changes in the training dataset, such as increasing pixel resolution and the number of input microstructural image frames. Expanding the range of driving force parameters, considering a wider range of materials, and incorporating various microstructural evolution phenomena in the training set will enhance the model's predictive capabilities in terms of applicability and generalizability.

In summary, the approach presented in this paper for predicting multi-generational microstructural evolution through Conv-LSTM neural networks, coupled with quantification and evaluation metrics, provides a valuable framework for advancing the understanding of material behavior, design, and optimization across a wide range of engineering applications.

6 Declarations

The authors declare to have no conflicts of interest that are relevant to the content of this article.

7 Acknowledgment

This work was supported by the National Science Centre, Poland (UMO-2021/42/E/ST5/00339). N.M. acknowledges the support by the HORIZON EUROPE European Research Council (101123107- μ TWIN). The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government.

8 Data Availability

The codes used in this work are available at <https://github.com/subediupadesh/Micro-Interface-ConvLSTM>.

References

- [1] Moelans, N., Blanpain, B., Wollants, P.: An introduction to phase-field modeling of microstructure evolution. *Calphad: Computer Coupling of Phase Diagrams and Thermochemistry* **32**(2), 268–294 (2008) <https://doi.org/10.1016/j.calphad.2007.11.003>
- [2] Prawoto, Y., Ikeda, M., Manville, S.K., Nishikawa, A.: Design and failure modes of automotive suspension springs. *Engineering Failure Analysis* **15**(8), 1155–1174 (2008) <https://doi.org/10.1016/j.engfailanal.2007.11.003>
- [3] Dallago, M., Winiarski, B., Zanini, F., Carmignato, S., Benedetti, M.: On the effect of geometrical imperfections and defects on the fatigue strength of cellular lattice structures additively manufactured via Selective Laser Melting. *International Journal of Fatigue* **124**, 348–360 (2019) <https://doi.org/10.1016/j.ijfatigue.2019.03.019>
- [4] Li, F., Li, J., Huang, T., Kou, H., Zhou, L.: Compression fatigue behavior and failure mechanism of porous titanium for biomedical applications. *Journal of the Mechanical Behavior of Biomedical Materials* **65**, 814–823 (2017) <https://doi.org/10.1016/j.jmbbm.2016.09.035>
- [5] Aksakal, B., Yildirim, O.S., Gul, H.: Metallurgical failure analysis of various implant materials used in orthopedic applications. *Journal of Failure Analysis and Prevention* **4**(3), 17–23 (2004) <https://doi.org/10.1007/s11668-996-0007-9>
- [6] Moghadasi, K., Mohd Isa, M.S., Ariffin, M.A., Mohd jamil, M.Z., Raja, S., Wu, B., Yamani, M., Bin Muhamad, M.R., Yusof, F., Jamaludin, M.F., Ab Karim, M.S., Abdul Razak, B., Yusoff, N.: A review on biomedical implant materials and

the effect of friction stir based techniques on their mechanical and tribological properties. *Journal of Materials Research and Technology* **17**, 1054–1121 (2022) <https://doi.org/10.1016/j.jmrt.2022.01.050>

- [7] Mohammadi, A., Edalati, P., Arita, M., Bae, J.W., Kim, H.S., Edalati, K.: Microstructure and defect effects on strength and hydrogen embrittlement of high-entropy alloy CrMnFeCoNi processed by high-pressure torsion. *Materials Science and Engineering A* **844**, 143179 (2022) <https://doi.org/10.1016/j.msea.2022.143179>
- [8] Krill, C.E., Chen, L.Q.: Computer simulation of 3-D grain growth using a phase-field model. *Acta Materialia* **50**(12), 3057–3073 (2002) [https://doi.org/10.1016/s1359-6454\(02\)00084-8](https://doi.org/10.1016/s1359-6454(02)00084-8)
- [9] Spatschek, R., Brener, E., Karma, A.: Phase field modeling of crack propagation. *Philosophical Magazine* **91**(1), 75–95 (2011) <https://doi.org/10.1080/14786431003773015>
- [10] Lo, Y.S., Borden, M.J., Ravi-Chandar, K., Landis, C.M.: A phase-field model for fatigue crack growth. *Journal of the Mechanics and Physics of Solids* **132**, 103684 (2019) <https://doi.org/10.1016/j.jmps.2019.103684>
- [11] Lowengrub, J.S., Ratz, A., Voigt, A.: Phase-field modeling of the dynamics of multicomponent vesicles: Spinodal decomposition, coarsening, budding, and fission. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **79**(3), 031926 (2009) <https://doi.org/10.1103/PhysRevE.79.031926>
- [12] Yang, M., Wang, L., Yan, W.: Phase-field modeling of grain evolutions in additive manufacturing from nucleation, growth, to coarsening. *npj Computational Materials* **7**(1), 56 (2021) <https://doi.org/10.1038/s41524-021-00524-6>
- [13] Subedi, U., Coutinho, Y.A., Malla, P.B., Gyanwali, K., Kunwar, A.: Automatic Featurization Aided Data-Driven Method for Estimating the Presence of Intermetallic Phase in Multi-Principal Element Alloys. *Metals* **12**(6), 964 (2022) <https://doi.org/10.3390/met12060964>
- [14] Subedi, U., Poudel, S., Gyanwali, K., Amorim Coutinho, Y., Matula, G., Kunwar, A.: State-of-the-Art Review on the Aspects of Martensitic Alloys Studied via Machine Learning. *Metals* **12**(11), 1884 (2022) <https://doi.org/10.3390/met12111884>
- [15] Vivek Bhave, C., Zheng, G., Sridharan, K., Schwen, D., Tonks, M.R.: An electrochemical mesoscale tool for modeling the corrosion of structural alloys by molten salt. *Journal of Nuclear Materials* **574**, 154147 (2023) <https://doi.org/10.1016/j.jnucmat.2022.154147>
- [16] Yang, H., Bao, R., Zhang, J., Peng, L., Fei, B.: Creep-fatigue crack growth

- behaviour of a nickel-based powder metallurgy superalloy under high temperature. *Engineering Failure Analysis* **18**(3), 1058–1066 (2011) <https://doi.org/10.1016/j.engfailanal.2010.12.025>
- [17] Chevalier, J., Gremillard, L., Deville, S.: Low-temperature degradation of zirconia and implications for biomedical implants. *Annual Review of Materials Research* **37**(1), 1–32 (2007) <https://doi.org/10.1146/annurev.matsci.37.052506.084250>
 - [18] Challa, V.S.A., Nune, K.C., Gong, N., Misra, R.D.K.: The significant impact of mechanically-induced phase transformation on cellular functionality of biomedical austenitic stainless steel. *Journal of the Mechanical Behavior of Biomedical Materials* **108**, 103815 (2020) <https://doi.org/10.1016/j.jmbbm.2020.103815>
 - [19] Oca Zapiain, D., Stewart, J.A., Dingreville, R.: Accelerating phase-field-based microstructure evolution predictions via surrogate models trained by machine learning methods. *npj Computational Materials* **7**(1), 3 (2021) <https://doi.org/10.1038/s41524-020-00471-8>
 - [20] Yang, K., Cao, Y., Zhang, Y., Fan, S., Tang, M., Aberg, D., Sadigh, B., Zhou, F.: Self-supervised learning and prediction of microstructure evolution with convolutional recurrent neural networks. *Patterns* **2**(5), 100243 (2021) <https://doi.org/10.1016/j.patter.2021.100243>
 - [21] Hu, C., Martin, S., Dingreville, R.: Accelerating phase-field predictions via recurrent neural networks learning the microstructure evolution in latent space. *Computer Methods in Applied Mechanics and Engineering* **397**, 115128 (2022) <https://doi.org/10.1016/j.cma.2022.115128>
 - [22] Kazemzadeh Farizhandi, A.A., Mamivand, M.: Spatiotemporal prediction of microstructure evolution with predictive recurrent neural network. *Computational Materials Science* **223**, 112110 (2023) <https://doi.org/10.1016/j.commatsci.2023.112110>
 - [23] Ahmad, O., Kumar, N., Mukherjee, R., Bhowmick, S.: Accelerating microstructure modelling via machine learning: a new method combining Autoencoder and ConvLSTM (2023) [arXiv:2305.00938](https://arxiv.org/abs/2305.00938)
 - [24] Gaston, D., Newman, C., Hansen, G., Lebrun-Grandié, D.: MOOSE: A parallel computational framework for coupled systems of nonlinear equations. *Nuclear Engineering and Design* **239**(10), 1768–1778 (2009) <https://doi.org/10.1016/j.nucengdes.2009.05.021>
 - [25] Tonks, M.R., Gaston, D., Millett, P.C., Andrs, D., Talbot, P.: An object-oriented finite element framework for multiphysics phase field simulations. *Computational Materials Science* **51**(1), 20–29 (2012) <https://doi.org/10.1016/j.commatsci.2011.07.028>

- [26] Lindsay, A.D., Gaston, D.R., Permann, C.J., Miller, J.M., Andrs, D., Slaughter, A.E., Kong, F., Hansel, J., Carlsen, R.W., Icenhour, C., Harbour, L., Giudicelli, G.L., Stogner, R.H., German, P., Badger, J., Biswas, S., Chapuis, L., Green, C., Hales, J., Hu, T., Jiang, W., Jung, Y.S., Matthews, C., Miao, Y., Novak, A., Peterson, J.W., Prince, Z.M., Rovinelli, A., Schunert, S., Schwen, D., Spencer, B.W., Veeraraghavan, S., Recuero, A., Yushu, D., Wang, Y., Wilkins, A., Wong, C.: 2.0 - moose: Enabling massively parallel multiphysics simulation. *SoftwareX* **20**, 101202 (2022) <https://doi.org/10.1016/j.softx.2022.101202>
- [27] Aagesen, L.K., Gao, Y., Schwen, D., Ahmed, K.: Grand-potential-based phase-field model for multiple phases, grains, and chemical components. *Physical Review E* **98**(2) (2018) <https://doi.org/10.1103/PhysRevE.98.023309>
- [28] Iquebal, A.S., Wu, P., Sarfraz, A., Ankit, K.: Emulating the evolution of phase separating microstructures using low-dimensional tensor decomposition and nonlinear regression. *MRS Bulletin* (2023) <https://doi.org/10.1557/s43577-022-00443-x>
- [29] Iwamatsu, M.: Direct numerical simulation of homogeneous nucleation and growth in a phase-field model using cell dynamics method. *The Journal of Chemical Physics* **128**(8) (2008) <https://doi.org/10.1063/1.2883652>
- [30] Wu, P., Iquebal, A.S., Ankit, K.: Emulating microstructural evolution during spinodal decomposition using a tensor decomposed convolutional and recurrent neural network. *Computational Materials Science* **224**, 112187 (2023) <https://doi.org/10.1016/j.commatsci.2023.112187>
- [31] Zhang, L., Tonks, M.R., Gaston, D., Peterson, J.W., Andrs, D., Millett, P.C., Biner, B.S.: A quantitative comparison between and elements for solving the Cahn–Hilliard equation. *Journal of Computational Physics* **236**, 74–80 (2013) <https://doi.org/10.1016/j.jcp.2012.12.001>
- [32] Nolte, D.D.: Introduction to Modern Dynamics, p. 512. Oxford University Press, ??? (2019). <https://doi.org/10.1093/oso/9780198844624.001.0001>
- [33] Yang, X., Ju, L.: Linear and unconditionally energy stable schemes for the binary fluid–surfactant phase field model. *Computer Methods in Applied Mechanics and Engineering* **318**, 1005–1029 (2017) <https://doi.org/10.1016/j.cma.2017.02.011>
- [34] Yang, X., Xiao, J.: On Linear and Unconditionally Energy Stable Algorithms for Variable Mobility Cahn-Hilliard Type Equation with Logarithmic Flory-Huggins Potential. *Communications in Computational Physics* **25**(3) (2019) <https://doi.org/10.4208/cicp.OA-2017-0259>
- [35] Binder, K.: Double-well thermodynamic potentials and spinodal curves: how real are they? *Philosophical Magazine Letters* **87**(11), 799–811 (2007) <https://doi.org/10.1080/09500830701496560>

- [36] Wang, X., Kou, J., Cai, J.: Stabilized Energy Factorization Approach for Allen–Cahn Equation with Logarithmic Flory–Huggins Potential. *Journal of Scientific Computing* **82**(2), 25 (2020) <https://doi.org/10.1007/s10915-020-01127-x>
- [37] Hussain, M.M., Ma, H., Huang, M., Gao, Z., Cao, J., Wang, C., Dong, C., Wang, Y., Kunwar, A.: Fabrication of cerium myristate coating for a mechanochemically robust modifier-free superwettability system to enhance the corrosion resistance on 316L steel by one-step electrodeposition. *Surface and Coatings Technology* **398**, 125970 (2020) <https://doi.org/10.1016/j.surfcoat.2020.125970>
- [38] Kunwar, A., Tonks, M.R., Shang, S., Song, X., Wang, Y., Ma, H.: Quantitative polynomial free energy based phase field model for void motion and evolution in Sn under thermal gradient. In: 2017 18th International Conference on Electronic Packaging Technology (ICEPT), pp. 1502–1507. IEEE, ??? (2017). <https://doi.org/10.1109/ICEPT.2017.8046720> . <http://ieeexplore.ieee.org/document/8046720/>
- [39] Bandyopadhyay, A.K., Ray, P.C., Gopalan, V.: Dynamical systems analysis for polarization in ferroelectrics. *Journal of Applied Physics* **100**(11) (2006) <https://doi.org/10.1063/1.2388124>
- [40] Shin, W., Lee, J., Chang, K.: The Effects of Inhomogeneous Elasticity and Dislocation on Thermodynamics and the Kinetics of the Spinodal Decomposition of a Fe-Cr System: A Phase-Field Study. *Metals* **10**(9), 1209 (2020) <https://doi.org/10.3390/met10091209>
- [41] Ahrens, J., Geveci, B., Law, C.: ParaView: An End-User Tool for Large Data Visualization ParaViewWeb View project. In: Los Alamos National Laboratory vol. 836. Elesvier, ??? (2005). <https://datascience.dsscale.org/wp-content/uploads/2016/06/ParaView.pdf>
- [42] Singhal, S.P., Herman, H., Hirvonen, J.K.: Spinodal decomposition in amorphous Au-implanted Pt. *Applied Physics Letters* **32**(1), 25–26 (1978) <https://doi.org/10.1063/1.89819>
- [43] Lecun, Y., Bengio, Y.: In: Arbib, M.A. (ed.) Convolutional Networks for Images, Speech and Time Series, pp. 255–258. The MIT Press, ??? (1995)
- [44] Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Computation* **9**, 1735–1780 (1997) <https://doi.org/10.1162/neco.1997.9.8.1735>
- [45] Olah, C.: Understanding LSTM Networks (2015). <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>
- [46] Elsworth, S., Guttel, S.: Time series forecasting using lstm networks: A symbolic approach (2020) [arXiv:2003.05672](https://arxiv.org/abs/2003.05672)

- [47] Bird, J.J., Faria, D.R., Ekart, A., Premebida, C., Ayrosa, P.P.S.: Lstm and gpt-2 synthetic speech transfer learning for speaker recognition to overcome data scarcity (2020) [arXiv:2007.00659](https://arxiv.org/abs/2007.00659)
- [48] Singh, S.P., Wang, L., Gupta, S., Gulyas, B., Padmanabhan, P.: Shallow 3D CNN for Detecting Acute Brain Hemorrhage from Medical Imaging Sensors. IEEE Sensors Journal **21**(13), 14290–14299 (2021) <https://doi.org/10.1109/JSEN.2020.3023471>
- [49] Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D.D., Chen, M.: Medical image classification with convolutional neural network. In: 2014 13th International Conference on Control Automation Robotics and Vision, ICARCV 2014, pp. 844–848. IEEE, ??? (2014). <https://doi.org/10.1109/ICARCV.2014.7064414> . <http://ieeexplore.ieee.org/document/7064414/>
- [50] Shi, X., Chen, Z., Wang, H., Yeung, D.Y., Wong, W.K., Woo, W.C.: Convolutional lstm network: A machine learning approach for precipitation nowcasting. Advances in Neural Information Processing Systems **2015-January**, 802–810 (2015) [arXiv:1506.04214](https://arxiv.org/abs/1506.04214)
- [51] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems **2017-December**, 5999–6009 (2017) [arXiv:1706.03762](https://arxiv.org/abs/1706.03762)
- [52] Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving Language Understanding by Generative Pre-Training. In: Homology, Homotopy and Applications, vol. 9, pp. 399–438 (2018)
- [53] Chollet, F., Others: Keras. GitHub (2015). <https://github.com/fchollet/keras>
- [54] Keras: ConvLSTM2D layer (2022). https://keras.io/api/layers/recurrent_layers/conv_lstm2d/
- [55] Amogh, J., Keras: Next-Frame Video Prediction with Convolutional LSTMs (2021). https://keras.io/examples/vision/conv_lstm/ https://github.com/keras-team/keras-io/blob/master/examples/vision/conv_lstm.py
- [56] Kunwar, A., Malla, P.B., Sun, J., Qu, L., Ma, H.: Convolutional neural network model for synchrotron radiation imaging datasets to automatically detect interfacial microstructure: An in situ process monitoring tool during solar PV ribbon fabrication. Solar Energy **224**, 230–244 (2021) <https://doi.org/10.1016/j.solener.2021.06.006>
- [57] Um, K., Hu, X., Wang, B., Thuerey, N.: Spot the difference: Accuracy of numerical simulations via the human visual system. ACM Transactions on Applied Perception **18**, 1–15 (2021) <https://doi.org/10.1145/3449064>

- [58] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* **13**, 600–612 (2004) <https://doi.org/10.1109/TIP.2003.819861>
- [59] Silva, E.A., Panetta, K., Agaian, S.S.: Quantifying image similarity using measure of enhancement by entropy. <https://doi.org/10.1111/12.720087> **6579**, 219–230 (2007) <https://doi.org/10.1111/12.720087>
- [60] Koenderink, J.J., Doorn, A.J.: Surface shape and curvature scales. *Image and Vision Computing* **10**(8), 557–564 (1992) [https://doi.org/10.1016/0262-8856\(92\)90076-F](https://doi.org/10.1016/0262-8856(92)90076-F)
- [61] Bradski, G.: The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000)
- [62] Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T.: scikit-image: image processing in python. *PeerJ* **2**, 453 (2014) <https://doi.org/10.7717/peerj.453>
- [63] Khalel, A.: Sewar (2018). <https://github.com/andrewekhalel/sewar>
- [64] Kumar, J., Chen, F., Doermann, D.: Sharpness Estimation for Document and Scene Images (2019). <https://github.com/umang-singhal/pydom>
- [65] Hunter, J.D.: Matplotlib: A 2D graphics environment. *Computing in Science and Engineering* **9**(3), 90–95 (2007) <https://doi.org/10.1109/MCSE.2007.55>
- [66] Subedi, U., Moelans, N., Tański, T., Kunwar, A.: Rapid portabilization of elasto-chemical evolution data for dental Ti-Cr alloy microstructure through sparsification and tensor computation. *Scripta Materialia* **244**, 116027 (2024) <https://doi.org/10.1016/j.scriptamat.2024.116027>
- [67] Brunet, D., Vrscay, E.R., Wang, Z.: On the mathematical properties of the structural similarity index. *IEEE Transactions on Image Processing* **21**, 1488–1499 (2012) <https://doi.org/10.1109/TIP.2011.2173206>
- [68] Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T.: scikit-image: image processing in Python. *PeerJ* **2**, 453 (2014) <https://doi.org/10.7717/peerj.453>
- [69] Catarious, D.M., Baydush, A.H., Floyd, C.E.: Characterization of difference of Gaussian filters in the detection of mammographic regions. *Medical Physics* **33**(11), 4104–4114 (2006) <https://doi.org/10.1111/1.2358326>
- [70] Zhao, F., DeSilva, C.J.S.: Use of the Laplacian of Gaussian operator in prostate ultrasound image processing **2**, 812–815 (2002) <https://doi.org/10.1109/iembs.1998.745557>

- [71] Fotin, S.V., Yankelevitz, D.F., Henschke, C.I., Reeves, A.P.: A multiscale Laplacian of Gaussian (LoG) filtering approach to pulmonary nodule detection from whole-lung CT scans (2019) [arXiv:1907.08328](https://arxiv.org/abs/1907.08328)
- [72] Kong, H., Akakin, H.C., Sarma, S.E.: A generalized laplacian of gaussian filter for blob detection and its applications. *IEEE Transactions on Cybernetics* **43**(6), 1719–1733 (2013) <https://doi.org/10.1109/TCYB.2012.2228639>
- [73] Sotak, G.E., Boyer, K.L.: The Laplacian-of-Gaussian kernel: a formal analysis and design procedure for fast, accurate convolution and full-frame output. *Computer Vision, Graphics, and Image Processing* **48**(2), 147–189 (1989) [https://doi.org/10.1016/S0734-189X\(89\)80036-2](https://doi.org/10.1016/S0734-189X(89)80036-2)
- [74] Lindeberg, T.: Image Matching Using Generalized Scale-Space Interest Points. *Journal of Mathematical Imaging and Vision* **52**(1), 3–36 (2015) <https://doi.org/10.1007/s10851-014-0541-0>
- [75] Gunn, S.R.: On the discrete representation of the Laplacian of Gaussian. *Pattern Recognition* **32**(8), 1463–1472 (1999) [https://doi.org/10.1016/S0031-3203\(98\)00163-0](https://doi.org/10.1016/S0031-3203(98)00163-0)
- [76] Besl, P.J., Jain, R.C.: Invariant surface characteristics for 3D object recognition in range images. *Computer Vision, Graphics, and Image Processing* **33**(1), 33–80 (1986) [https://doi.org/10.1016/0734-189X\(86\)90220-3](https://doi.org/10.1016/0734-189X(86)90220-3)
- [77] Cantzler, H., Fisher, R.B.: Comparison of HK and SC curvature description methods. In: Proceedings Third International Conference on 3-D Digital Imaging and Modeling, pp. 285–291. IEEE Comput. Soc. <https://doi.org/10.1109/IM.2001.924458> . <http://ieeexplore.ieee.org/document/924458/>
- [78] Saha, T., Mookerjee, A.: A phase-stability study of PdRh alloys. *Journal of Physics Condensed Matter* **9**(10), 2179–2186 (1997) <https://doi.org/10.1088/0953-8984/9/10/007>
- [79] Sigala-Garcia, D.A., Lopez-Hirata, V.M., Saucedo-Munoz, M.L., Dorantes-Rosales, H.J., Villegas-Cardenas, J.D.: Phase-Field Simulation of Spinodal Decomposition in Mn-Cu Alloys. *Metals* **12**(7), 1220 (2022) <https://doi.org/10.3390/met12071220>

A Appendix: Parametrization of driving force G

To visualize the effects of coefficient changes on the landscape of Gibbs free energy, as governed by Equation 3, we initially selected a random set of parametric values as a reference, as presented in Table A1. In Equation 3, 'c' represents the composition of an element in the alloy system, while all other variables are coefficients with specific meanings that determine certain characteristics of the Gibbs free energy curve. In the following sections, we individually varied each of these coefficients and visualized their effects.

Table A1 Values of six coefficients taken as initial reference

α (J/mol)	β (J/mol)	γ (J/mol)	δ (J/mol)	ϵ	c_{eq}
-5.5e4	0	2.75e5	-0.5e3	1.5	0.65

A.1 Effects of change in α

The coefficient α determines the amplitude of the double-well structure in the Gibbs free energy curve. To create minima at two points and promote phase separation, a negative value of α is necessary. This separation leads to the formation of distinct phases. As illustrated in Figure A.1a, a higher α value of $-5.5e4 J/mol$ causes the red curve to dip more than the green curve.

A.2 Variation in β

The β parameter is associated with the slope of the free energy curve, influencing the stability of separated phases. When $\beta = 0$, both separated phases are equally preferred, resulting in a symmetrical curve. However, a non-zero β value favors one of the phases, making the free energy curve asymmetrical. Figure 1a illustrates that at composition ($c = 0.533$), the concave region of the curve is more prominent than at composition ($c = 0.238$). Consequently, the free energy is lowest at this composition, indicating a non-zero β value ($\beta = -0.4e4 J/mol$) and a preference for the phase at ($c = 0.533$). In Figure A.1b, the three different curves representing variations in the driving force also have different β values. The green curve has a positive β value of $5.0e3 J/mol$, the magenta curve has a negative β value ($-5.0e3 J/mol$), while the dotted black curve is completely symmetric, with a slope of 0, indicating $\beta = 0$. This type of expression is used in [31]. In summary, the β coefficient in Equation 3 can significantly affect the shape of a free energy curve.

A.3 Changing γ

The coefficient γ introduces a higher-order interaction between the two components of a binary alloy. Similar to α , it is associated with the quadratic term of the expression, affecting the depth of the double-well structure, as shown in Figure A.1c. For a binary

alloy system to undergo phase separation through phase decomposition, it is necessary for γ to have an opposite sign to that of α . In a thermodynamically and physically valid Gibbs free energy of a system, the value of α must always be negative, making γ positive. A positive γ value facilitates phase separation, while an opposite value hinders the separation process, reducing the double-well shape to an inverted parabola.

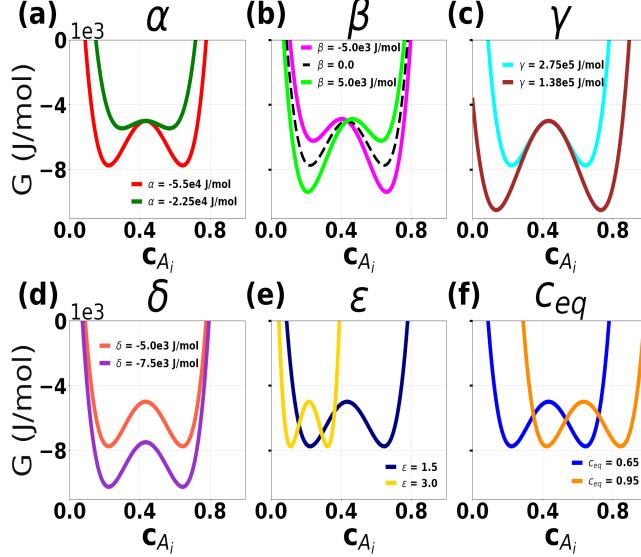


Fig. A.1 Gibbs free energy landscape fluctuation with the variation of parameters- (a) α , (b) β , (c) γ , (d) δ , (e) ϵ , and (f) c_{eq}

A.4 Role of δ

The term δ is independent of composition but plays a major role in accounting for external factors related to the enthalpy and entropy of the system. It represents a global energy offset that does not alter the shape of the energy function or affect the relative stability or nature of the double-wells. However, variations in δ cause the entire curve to shift vertically, uniformly impacting the overall energy level of the system. This effect is illustrated in Figure A.1d, where the magnitude of δ determines the vertical position of the free energy curve.

A.5 Parameter ϵ

The unit-less term ϵ in the equation is associated with the characteristic range of the composition, its value determines the width of double-well curvature. As shown in Figure A.1e a larger value of ϵ (yellow curve with $\epsilon = 3$) means the phase decomposition is observed in narrower compositional range whereas for a smaller value (dark blue curve with $\epsilon = 1.5$) the decomposition occurs over a broader compositional range. For

a realistic phenomenon with non-negative compositional values, ϵ should always be greater than zero.

A.6 c_{eq} in G

The c_{eq} parameter is used as a reference or origin point for the energy function, the difference between the composition c and c_{eq} affects the double-well and energy landscape's position, it shifts the horizontal position of the curve without changing the shape and symmetry of the free energy landscape as shown in Figure A.1f.

A.7 Superposed effect

Collectively, all the coefficients and parameters of free energy expression defines the curvature, slope, polynomial behavior and global energy offset of the system, as a whole they determine the characteristic nature of the landscape of energy function and drive the phase separation process.

B Appendix: Prediction Accuracy Beyond GEN-3

After Generation 3, the model's predictions become completely independent of the input frames supplied from the phase field simulations, relying solely on the results of previous predictions, thus entering the 'prediction-based prediction' zone. Our analysis reveals that for model architecture A2 with 3 frames as input maintains a microstructural similarity index of 90% or higher up to the 10th generation, as shown in Figure B.1, with an estimated computational time of 273.25 seconds. However, after the 10th generation, the accuracy of the model, measured by the microstructural similarity index, falls below the 90% mark. Consequently, we do not recommend using predictions beyond this generation for critical decision-making processes in any kind of failure analysis.

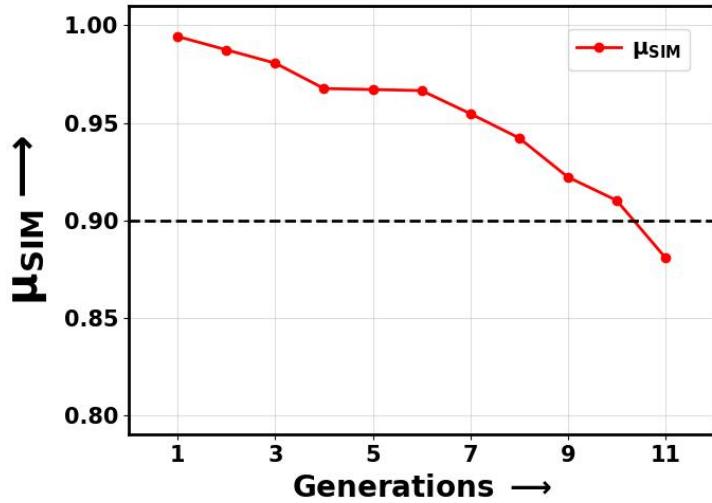


Fig. B.1 Microstructural similarity index (μ_{SIM}) of predicted consecutive 10 generations for model architecture A2 with 3 frames as input to the model. It is observed that the prediction similarity index is above 90% for the 10 consecutive generations and only after the 10th generation the prediction similarity index (accuracy) drops the 90% bench-mark.