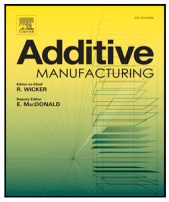
Additive Manufacturing 66 (2023) 103461

Contents lists available at ScienceDirect

Additive Manufacturing

journal homepage: www.elsevier.com/locate/addma



Research paper

Droplet evolution prediction in material jetting via tensor time series analysis

Luis Javier Segura [a], Zebin Li [b], Chi Zhou [b], Hongyue Sun [b,*]

[a] Department of Industrial Engineering, University of Louisville, Louisville, KY 40292, United States
[b] Department of Industrial and Systems Engineering, University at Buffalo, SUNY Buffalo, NY 14260, United States

ARTICLE INFO

Keywords:
Material jetting
Additive manufacturing
Droplet evolution
Tensor time series
Deep learning

ABSTRACT

Material Jetting (MJ) process is an additive manufacturing process that is able to produce structures with high resolutions. The performance and quality of the MJ printed parts extensively rely on the droplet morphology and behavior. However, obtaining consistent and stable droplet morphology and behavior is difficult because the droplets are very sensitive to different material and process parameters. Testing/studying all these parameter combinations is time-consuming due to the high-experimental and high-computational costs. We thus study the prediction of droplet behaviors under different material and process parameters. To achieve this, we propose to leverage the underlying relationships shared across droplet evolution behaviors with diverse material and process parameters (referred to as "cross-linked" relationship hereafter) as well as the spatial–temporal relationships of droplet evolution, and capture these with the Network of Tensor Time Series ($N_E T^3$). The distinct droplet behaviors are regarded as co-evolving time series (i.e., Tensor Time Series (TTS)) since they share the same physics principles. In particular, we capture the cross-linked and spatial relationships of TTS by the Tensor Graph Convolutional Network (TGCN), and capture the temporal relationship of TTS by the Tensor Recurrent Neural Network (TRNN), respectively. The features from TGCN and TRNN are passed to Multilayer Perceptron (MLP) for predicting future droplet behaviors. The proposed methodology is demonstrated with simulated (i.e., physics-based models) and experimental (i.e., MJ printing observation with vision system) droplet evolution videos and is able to accurately and efficiently make predictions for seen (i.e., forecasting for a future time) and unseen (i.e., new droplet evolution sequence prediction) material/process parameters.

1. Introduction

Additive Manufacturing (AM) processes are able to produce intricate parts and products that were not possible with traditional manufacturing processes [1,2]. Material jetting, i.e., Inkjet Printing (IJP), has been able to produce functional parts with complex structures [3] for electronics [4], energy [5], biomedical [6], etc., applications. For instance, high-performance transistor arrays and circuits [7], electrodes for microbatteries [8], scaffolds [9], have been successfully tested.

The IJP process (see Fig. 1), a material jetting-based AM technique, jets droplets of inks/solutions either continuously or in a Drop-On-Demand (DOD) fashion on the top of a substrate to additively create a part. In particular, the Piezo-electric Inkjet (PIJ) printing technology accurately supplies droplets in DOD mode [10] and is shown in Figs. 1 (a–b). Specifically, the individual droplet formation mechanism (i.e., droplet evolution) extensively determines the process performance and product quality. However, having consistent and stable droplets is difficult to achieve since they are prone to instabilities due to process parameters (e.g., electrical waveform, back-pressure), material properties (e.g., density, viscosity), and ambient conditions (e.g., temperature, humidity) variations [11]. Fig. 1(c) shows different simulated (Fig. 1 (c.1)) and experimental (Fig. 1 (c.2)) droplet evolution behaviors. Thus, achieving consistent droplet behaviors becomes paramount. The printing process success or failure has been studied by classifying droplet ejection into single spherical droplet, satellite droplets, etc. [12]; nevertheless, understanding the full droplet evolution in the IJP process is crucial to obtain droplet with desirable characteristics, such as, volume, velocity, and shape, and to perform process analysis [13].

Traditionally, researchers have studied droplet evolution behaviors experimentally/empirically and numerically [14,15]. With the advent of advanced sensing technology, specifically vision systems (e.g., Charged Coupled Device (CCD) and high-speed cameras), experimental approaches have been possible to study the IJP droplet behaviors [16,17]. For instance, several researchers have focused on analyzing the influence of process parameters and ink/solution properties on piezoelectric inkjet droplet formation by using cameras coupled with amplification systems [18,19]. In addition, vision systems support the analysis of empirical dimensionless numbers, namely Ohnesorge (Oh), Weber (We), and Reynolds (Re) numbers [20]. These numbers are

* Corresponding author.
E-mail address: hongyues@buffalo.edu (H. Sun).

https://doi.org/10.1016/j.addma.2023.103461
Received 11 August 2022; Received in revised form 29 January 2023; Accepted 19 February 2023
Available online 24 February 2023
2214-8604/Published by Elsevier B.V.

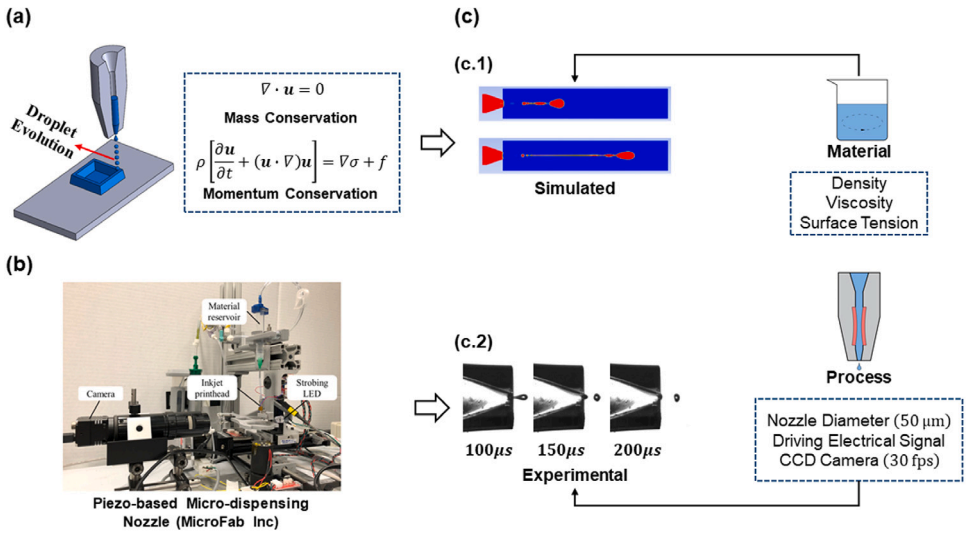


Fig. 1. An Illustration of the Inkjet Printing (IJP) Process: (a) Piezoelectric inkjet (PIJ) printing process, (b) PIJ printing experimental setup, and (c) Droplet behaviors: (c.1) simulated and (c.2) experimental.

mainly used to determine the jettability regime in the IJP process [21,22]. However, these numbers disregard the shape of the droplets (i.e., droplet evolution behaviors), which considerably influences the quality of the printed parts. These approaches utilize several snapshots to perform their analyses and performing such analyses for different material/process parameters can be time-consuming and expensive. In addition, these methods neglect the temporal relationship, which can be useful to make predictions of the droplet evolution during the printing process.

Numerical methods have also shown their potential for a better understanding of the droplet formation process [23]. Particular attention has been given to the Volume of Fluid (VOF) method [24] and lattice Boltzmann method [25] . For the first, the Navier–Stokes (NS) equations are solved via VOF method to study and evaluate the effect of operating parameters (i.e., material properties) on the droplet formation in the IJP process [23,24]. The mesoscopic kinetic equations are considered to simulate the droplet formation and coalescence processes in microfluidics [25,26]. These approaches have shown good agreement with reality; however, they lack fluid flow pattern identification, which makes the droplet evolution forecasting and prediction for new material/process parameters difficult.

In this paper, we focus on the single droplet behavior prediction for both seen (i.e., forecasting for a future time) and unseen (i.e., new droplet evolution sequence prediction) material/process parameters. Generally, droplet evolution prediction is hard to achieve in the IJP process [27]. In particular, the droplet evolution phenomenon can be regarded as a time series with spatial–temporal relationships [27]. Here, the spatial relationship refers to relationship of pixels within a video frame; while the temporal relationship refers to the relationship in pixels from one frame to other frames (see Fig. 1 (c)). Several methods, such as Autoregressive Integrated Moving Average (ARIMA) and Kalman filtering, have been widely used to forecast time series [28,29]. Nevertheless, these approaches do not capture the complex non-linear spatial–temporal relationships in the droplet evolution. The emergence of Deep Learning (DL) has allowed to capture complex and hidden information from massive data (e.g., droplet evolution videos) [30]. For instance, Convolutional Neural Networks (CNNs) have been used to predict the velocity field around a cylinder using the pressure field on the cylinder [31]. In addition, deep CNN and Multilayer Perceptron (MLP) have been deployed to forecast incompressible laminar steady flow field over airfoils [32]. Nonetheless, most of the existing DL architectures are mainly used for dimensionality reduction or current time flow reconstruction, which are effective to focus mainly on sequence learning or spatial learning [33], but fail to model the spatial–temporal dynamics simultaneously.

Recently, DL architectures able to capture spatial–temporal relationships have been studied [34,35]. An attempt to simultaneously model the spatial–temporal droplet dynamics in the PIJ printing process is presented in [27]. In particular, an unsupervised learning method is deployed for the droplet evolution prediction using a deep recurrent neural network. However, it is not able to capture the cross-linked relationships (i.e., shared relationships across different droplet behaviors generated with the same forming mechanism) among the droplet evolution behaviors at various material/process parameters, and requires large datasets (e.g., 4500 droplet sequences were used in [27]).

The objective of this paper is to forecast and predict the droplet evolution in the PIJ printing process by capturing the cross-linked relationships of different material/process parameters and spatial–temporal relationships of droplet evolution. Here, various droplet evolution behaviors (see Fig. 1(c)) from either simulated or experimental data under diverse material/process parameters can be regarded as co-evolving time series (i.e., Tensor Time Series (TTS)) since they share the same forming mechanism. The main contributions of this paper are as follow: (a) The cross-linked relationships of the co-evolving time series are simultaneously captured by Tensor Graph Convolutional Network (TGCN) and Tensor Recurrent Neural Network (TRNN) in Network of Tensor Time Series ($\mathrm{N_eT^3}$) [36]. The TGCN is in charge of capturing the spatial relationship within and cross-linked relationship among the co-evolving time series. The TRNN captures the temporal dynamics of the different droplet evolution behaviors in the TTS. (b) Predicting droplet behavior sequences for unseen material/process parameters is feasible by exploiting the synergy among different networks. To our best knowledge, this is the first study dedicated to modeling the droplet evolution forming mechanism in the PIJ printing process that systematically capture the cross-linked, spatial and temporal relationships to predict the unseen and seen material/process parameters. An overall prediction evaluation was performed by means of Structural Similarity (SSIM) and Mean Square Error (MSE) indexes.

The paper is organized as follows. Section 2 will briefly discuss related studies on data-driven modeling for IJP and time series analysis. The deployed methodology will be presented in Section 3. Section 4 will show the results. Finally, Section 5 will conclude the paper and discuss the future work.

2. Literature review

2.1. Data-driven modeling for inkjet printing

Several Machine Learning (ML) methodologies have been explored in IJP. For example, Wu et al. [37] proposed an ensemble learning method which combined random forest, least absolute shrinkage and selection operator, eXtreme Gradient Boosting (XGBoost) and Support Vector Regression (SVR) to predict droplet velocity and volume during the droplet formation process of inkjet-based bioprinting. Pan et al. [38] modeled the effects of waveform voltage amplitude on droplet deposition errors, droplet volume and droplet velocity during IJP process by using Kriging, radial basis function and SVR. Stoyanov et al. [39] developed a state-space model to predict the thickness of printed products and realized model predictive control based on the captured dynamic behavior of the IJP process. Other similar studies can be seen in [40,41].

Due to the powerful capability of dealing with non-linear relationships, applying DL techniques for IJP process modeling is getting researchers' attention. For instance, Tourloukis et al. [42] proposed a nonlinear autoregressive neural network model to forecast the layer thickness of inkjet-printed products. Inyang-Udoh et al. [43] proposed a Recurrent Neural Network (RNN) to forecast the layer height profiles. The RNN incorporated a physical-based model to reduce the large number of required data for model training. Based on the proposed model, a feedback control of IJP process was further developed. Shi et al. [44] tried to address the challenges of inkjet-based bioprinting, such as satellite generating, too-large droplet diameter, and too-low droplet velocity by using fully connected neural networks. Ogunsanya et al. [45] developed a video-based in-situ IJP monitoring framework to classify the droplet modes (i.e., normal, satellite and no-droplet) by using a back-propagation neural network, and the satellite droplets, droplets aspect ratio, the size of droplet as well as the droplet velocity were extracted from video frames as input features. More examples can be seen in [46,47]

Even though the aforementioned approaches have shown to be effective at improving the IJP process by analyzing single droplet dynamic behaviors (e.g., droplet velocity), they do not exploit the synergy of co-evolving behaviors to make predictions of unseen material/process parameters.

2.2. Time series analysis

As mentioned before, the droplet evolution can be considered as a time series with spatial–temporal relationships. We thus review the methods in time series analysis here. Time series analysis aims at discovering the patterns of data collected over time. Conventional time series analysis methods include autoregressive, moving-average, autoregressive moving-average, ARIMA, etc. These methods cannot capture the complex non-linear relationships that are present in the IJP process. Recently, the rising of ML techniques enable the modeling of complex time series data. By considering the time series forecasting problems as supervised learning problems, several ML techniques, such as SVR, Gaussian Process, have been applied in the time series analysis [48–50]. In addition, DL techniques are also applied to deal with time series data, in order to handle non-linear relationships. Various neural network architectures have been proposed to model different time series data across different domains. For instance, seriesNet [51], WSAEs-LSTM [52], and RCLSTM [53]. [54,55] give good summaries on applying DL techniques in time series analysis. We refer interested readers to [56,57]. Although ML and DL techniques are able to capture complex non-linear relationships, they often ignore the networked relationships among co-evolving time series.

It is worth to mention that the majority of the DL methods for time series analysis have focused on analyzing individual time series [27]; therefore, neglecting the benefits of capturing the synergy among co-evolving time series. Co-evolving time series structures (i.e., TTS) can be exploited to learn the cross-linked and spatial–temporal relationships of the time series [36]. In particular, Jing et al. [36] integrated the TGCN and TRNN in a novel model called N_ET^3 to jointly capture the relationships. In this paper, to investigate the droplet evolution among co-evolving time series, we model the simulated and experimental droplet videos based on the N_ET^3, and perform the evolution prediction for the seen and unseen material/process parameters.

3. Proposed framework

3.1. Overview of the proposed framework

A scheme of the proposed framework for the IJP droplet evolution prediction is presented in Fig. 2. In the printing process, the droplet formation is crucial towards the performance and quality of the printed parts. The droplet evolution behaviors substantially depend on the material properties and process settings. Computational Fluid Dynamics (CFD) and vision systems are used to study and capture, respectively, the different droplet behaviors. In particular, simulated and experimental data (i.e., droplet videos) are collected as shown in Fig. 2(a). Each droplet video is considered as a time series and different droplet videos are regarded as co-evolving time series because they share the same forming mechanism. Hence, data preparation is necessary to form a network of TTS (see Fig. 2(b)). Subsequently, the new data structure is analyzed via N_ET^3 [36], which deploys TGCN and TRNN to jointly capture the cross-linked and spatial–temporal relationships of the TTS (see Fig. 2(c)). After this, predictions for seen and unseen material/process parameters are possible (see Fig. 2(d)). Finally, the predicted droplet sequences are compared with the original sequences from the simulated and experimental droplet videos.

In the proposed method, the simulation data generated from the physics-based model and the training of N_ET^3 model are performed offline. The method is applied independently to the simulated and experimental data. Directly training a network by combining the experiment and simulation data, even from the same parameters, will not achieve a good performance. This is because the resolution discrepancy and morphology discrepancies between the simulations and experiments. Since the purpose of this study is to capture the cross-linked relationships and spatial–temporal relationships in material jetting, two separate networks are trained for the experimental and simulated data, respectively, to demonstrate the proposed framework. Once the models are trained, fast droplet evolution predictions for simulated and experimental video data can be achieved.

3.2. Simulation and experimental data acquisition

3.2.1. Simulated data collection

For the simulated videos, the droplet formation process is modeled by a CFD model. The Navier–Stokes equations govern the physical model mass and momentum conservation for the liquid–gas interface (see the modeled system in Fig. 1 (c.1)), and it is assumed that the fluids are viscous, axisymmetric, and incompressible. Details of the model formulation can be seen in our previous work [27].

The CFD model is solved in ANSYS-Fluent. Additionally, the VOF approach, which allows to track the shape and position of droplets, is utilized. We then generate simulated droplet videos with various material properties (i.e., density, viscosity, surface tension), such as water, glycerol, graphene oxide-based inks, shown in Table 1.

3.2.2. Experimental data collection

The hardware of our video collection system is shown in Fig. 1(b). The droplets are generated by a piezo-based micro-dispensing nozzle

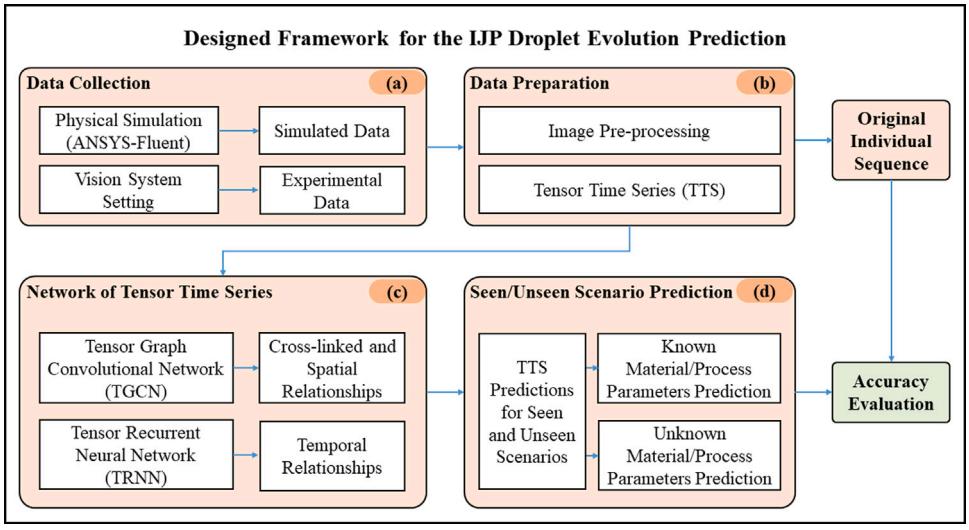


Fig. 2. An Illustration of the Proposed Methodology for the IJP Droplet Evolution Prediction: (a) Simulated and experimental data collection, (b) Data preparation, (c) Construction of the network of tensor time series, and (d) Droplet evolution predictions.

Table 1
Ranges of the material properties for simulated data collection.

Material properties	Low levels	High levels
Density (kg/m^3)	800	8000
Viscosity (kg/m.s)	0.0005	0.15
Surface tension (dyn/cm)	50	80

Table 2
Process settings ranges for experimental data collection.

Process settings	Low levels	High levels
Back-pressure (in H_2O)	−6	−3
Dwell time (μs)	28.2	35.8
Echo time (μs)	56.5	71.6
Dwell voltage (V)	48.1	70
Echo voltage (V)	−70	−48.1

(MicroFab Inc.). This print-head system is suitable for typical Newtonian materials, such as water and glycerol. The nozzle's diameter is 100 μm. The process settings for the experimental video generation are specified in Table 2. The droplet videos are captured by a CCD camera (Sensor Technologies Inc.) coupled with a magnification lens. The video resolution is 480 × 640 pixels. Finally, strobing technology is utilized to capture the droplet evolution behavior after the droplet is ejected, as shown in Fig. 1 (c.2).

3.3. Network of tensor time series

The droplet evolution behaviors (i.e., droplet time series) are crucial in the IJP process, and capturing their inherited spatial–temporal dynamics for droplet morphology predictions is challenging. This becomes even more evident when distinct droplet evolution behaviors (i.e., co-evolving time series) are considered. Co-evolving time series structures (i.e., TTS) can be exploited to capture the synergy among individual droplet behaviors corresponding to different material/process parameters by constructing graphs/networks [36]. In general, a network of tensor time series is composed of a tensor time series $\{\boldsymbol{S}_t\} \in \mathbb{R}^{N_1 \times \cdots \times N_M \times T}$ (i.e., co-evolving droplet videos in our case) and a set of adjacency matrices $A_m \in \mathbb{R}^{N_m \times N_m}$, where $m \in [1, \ldots, M]$ is associated with the non-temporal modes (i.e., material/process parameters and spatial location adjacency matrices, hence $M = 2$ in this study). Fig. 3 displays an illustration of the data structure in a 2nd mode network of

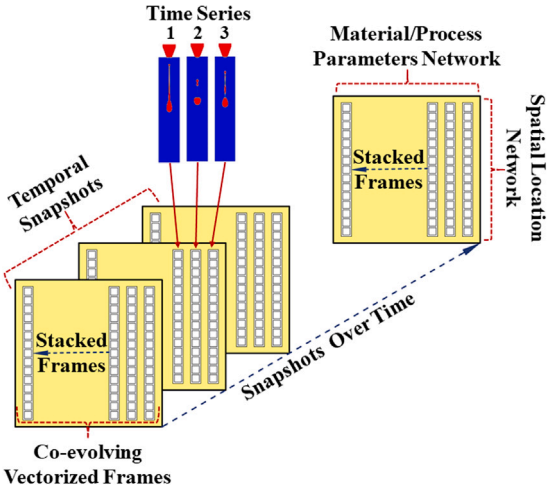


Fig. 3. A Schematic Illustration of a 2nd Mode Network of Tensor Time Series.

tensor time series. The data in Fig. 3 is used to learn the cross-linked and spatial–temporal relationships of the network of tensor time series in NeT^3 [36].

An overview of the NeT^3 is displayed in Fig. 4. Here, the cross-linked and spatial relationships, and temporal relationships are captured by the TGCN and TRNN modules, respectively. At every time step t, the tth snapshot $\boldsymbol{S}_t \in \mathbb{R}^{N_1 \times \cdots \times N_M}$ along with its adjacency matrices $\{A_m \in \mathbb{R}^{N_m \times N_m}\}_{m=1}^{M}$ are inputted to the TGCN to extract the node embedding tensor $\boldsymbol{\mathcal{H}}_t$ and encode the cross-linked and spatial relationships of each snapshot. After that, the $\boldsymbol{\mathcal{H}}_t$ tensor is used to encode the temporal dynamics via TRNN and produce the reconstructed hidden tensor $\boldsymbol{R}_t$. Finally, the output module takes the $\boldsymbol{\mathcal{H}}_t$ and $\boldsymbol{R}_t$ tensors to predict the future snapshot.

3.3.1. Tensor graph convolutional network

TGCN [36] extends Graph Convolutional Network (GCN) [58] for flat graphs to tensor graphs based on multi-dimensional convolution, so that the graphs for both material/process parameters and spatial location can be jointly captured. Given a snapshot $\boldsymbol{S}_t \in \mathbb{R}^{N_1 \times \cdots \times N_M}$

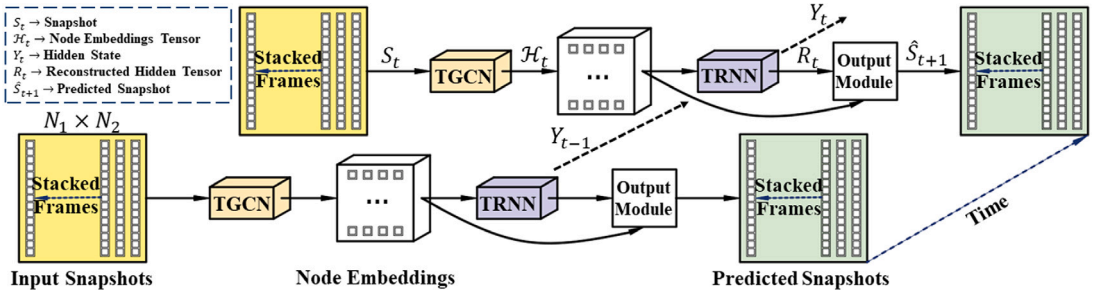


Fig. 4. A Scheme of the NeT[3] Applied to a 2nd Mode Network of Tensor Time Series.

(i.e., input snapshots in Fig. 4) and its adjacency matrices $\{A_m \in \mathbb{R}^{N_m \times N_m}\}_{m=1}^{M}$, the tensor graph Fourier transform is given by:

$$\tilde{\mathcal{S}}_t = \mathcal{S}_t \prod_{m=1}^{M} \times_m \boldsymbol{\Phi}_m \tag{1}$$

where $\boldsymbol{\Phi}_m$ is the eigenvector matrix of the graph Laplacian matrix $L_m = \boldsymbol{\Phi}_m \Lambda_m \boldsymbol{\Phi}_m^T$, here $\Lambda_m \in \mathbb{R}^{N_m \times N_m}$ is the matrix of eigenvalues for A_m, and $\times_m$ denotes the m-mode product. The spectral convolution for tensor graphs is defined as:

$$\mathcal{G}_t \star \tilde{\mathcal{S}}_t = \mathcal{S}_t \prod_{m=1}^{M} \times_m \boldsymbol{\Phi}_m^T diag(\tilde{g}_m) \boldsymbol{\Phi}_m \tag{2}$$

where $\mathcal{G}_t \in \mathbb{R}^{N_1 \times \cdots \times N_M}$ is a multi-dimensional filter, $\tilde{g}_m = \boldsymbol{\Phi}_m^T g_m$ is the Fourier transformed filter for the mth mode, and $\star$ is the convolution operation. To simplify the multi-dimensional spectral convolution operations and be able to parameterize the free parameters in the filters $\mathcal{G}_t$, Chebyshev approximation is deployed. In particular, this is given by approximating the Fourier transformed filters $\tilde{g}_m$ with Chebyshev polynomials.

$$\begin{aligned} \mathcal{G}_{\theta,t} \star \tilde{\mathcal{S}}_t &= \mathcal{S}_t \prod_{m=1}^{M} \times_m \boldsymbol{\Phi}_m^T \left(\sum_{p_m=0}^{P} \theta_{m,p_m} T_{p_m}(\tilde{\Lambda}_m) \right) \boldsymbol{\Phi}_m \\ &= \mathcal{S}_t \prod_{m=1}^{M} \times_m \boldsymbol{\Phi}_m^T \sum_{p_m=0}^{P} \theta_{m,p_m} T_{p_m}(\tilde{L}_m) \end{aligned} \tag{3}$$

where $\mathcal{G}_{\theta,t}$ denotes the convolution filter parameterized by θ, $T_{p_m}(x)$ is the Chebyshev polynomials defined by $T_{p_m}(x) = 2xT_{p_m-1}(x) - T_{p_m-2}(x)$ with $T_0(x) = 1$ and $T_1(x) = x$, $\tilde{\Lambda}_m$ is the normalized eigenvalues matrix, and p_m is the order of the polynomials. Notice that the same polynomials degree P are used for all the non-temporal modes (i.e., material/process parameters and spatial location in our case). Once spectral graph convolution on tensor graphs/networks is possible, the layer-wise updating function is established by exploiting the linearity of the m-mode product in Eq. (3); details of the derivation can be seen in [36], and is expressed as:

$$\mathcal{G}_{\theta,t} \star \tilde{\mathcal{S}}_t = \sum_{\exists p_m=1} \theta_{p_1,\ldots,p_M} \mathcal{S}_t \prod_{p_m=1} \times_m \tilde{A}_m + \theta_{0,\ldots,0} \mathcal{S}_t \tag{4}$$

where $\tilde{A}_m = D^{-\frac{1}{2}} A_m D^{-\frac{1}{2}}$ is the convolutional filter. Here, $p_m = [p_1, \ldots, p_M] \in [0,1]^M$ works as an indicator on whether applying $\tilde{A}_m$ to $\mathcal{S}_t$ or not. Finally, based on Eq. (4), the node embedding tensors $\mathcal{H}_t \in \mathbb{R}^{N_1 \times \cdots \times N_M \times d}$ that condense the cross-linked and spatial relationships of the input snapshots (see Fig. 4) can be computed as follows:

$$\begin{aligned} \mathcal{H}_t &= TGCN\left(\mathcal{S}_t, \{A_m\}_{m=1}^{M}\right) \\ &= \sigma\left(\sum_{\exists p_m=1} \mathcal{S}_t \prod_{p_m=1} \times_m \tilde{A}_m \times_{M+1} \Theta_{p_1,\ldots,p_M} + \mathcal{S}_t \times_{M+1} \Theta_0 \right) \end{aligned} \tag{5}$$

where Θ is the parameter matrix and $\sigma(\cdot)$ is activation function.

3.3.2. Tensor recurrent neural network

Once the spatial relationships are captured in $\mathcal{H}_t$ via TGCN, it is necessary to incorporate the temporal dynamics to $\mathcal{H}_t$ (see Fig. 4). TRNN is deployed for this purpose [36]. We first perform Tucker decomposition for $\mathcal{H}_t$ to reduce the dimension. After tensor decomposition, the temporal dynamics among the co-evolving time series (i.e., different droplet behaviors) are embedded into its core tensors: $\mathcal{Z}_t = \mathcal{H}_t \prod_{m=1}^{M} \times_m U_m^T$, where $U_m \in \mathbb{R}^{N'_m \times N_m}$ denotes the orthonormal parameter matrix. Here, $\mathcal{Z}_t \in \mathbb{R}^{N'_1 \times \cdots \times N'_m \times d}$ and $N'_m < N_m$. To achieve this, a Tensor Linear Layer (TLL) is defined as:

$$TLL(\mathcal{Z}_t) = \mathcal{Z}_t \prod_{m=1}^{M+1} \times_m W_m + b \tag{6}$$

where W_m is the linear transition parameter matrix and b is the bias vector. By replacing the linear functions in the Long Short-Term Memory (LSTM) architecture with the TLL, the updated functions for the TRNN are: $\mathcal{F}_t = \sigma\left(TLL_{fz}(\mathcal{Z}_t) + TLL_{fy}(\mathbf{Y}_{t-1})\right)$, $\mathcal{I}_t = \sigma\left(TLL_{iz}(\mathcal{Z}_t) + TLL_{iy}(\mathbf{Y}_{t-1})\right)$, $\mathcal{O}_t = \sigma\left(TLL_{oz}(\mathcal{Z}_t) + TLL_{oy}(\mathbf{Y}_{t-1})\right)$, $\tilde{C}_t = \tanh\left(TLL_{cz}(\mathcal{Z}_t) + TLL_{cy}(\mathbf{Y}_{t-1})\right)$, $C_t = \mathcal{F}_t \odot C_{t-1} + \mathcal{I}_t \odot \tilde{C}_t$, and $\mathbf{Y}_t = \mathcal{O}_t \odot \sigma(C_t)$, where $\mathbf{Y}_t \in \mathbb{R}^{N'_1 \times \cdots \times N'_M \times d'}$ is the hidden state tensor at the time step t; $\mathcal{F}_t$, $\mathcal{I}_t$, and $\mathcal{O}_t$ are the forget gate, input gate, and output gate, respectively; $\tilde{C}_t$ is the tensor that updates the cell memory C_t; tanh denotes the tangent activation function, and $\odot$ is the Hadamard product.

To make predictions of each time series (i.e., different droplet evolution behaviors), tensor reconstruction is performed via $\mathcal{R}_t = \mathbf{Y}_t \prod_{m=1}^{M} \times_m U_m$ so that the original tensor dimensions are obtained (i.e., $\mathcal{R}_t \in \mathbb{R}^{N_1 \times \cdots \times N_m \times d}$). The temporal dynamics are encoded in the reconstructed tensor $\mathcal{R}_t$.

Once the spatial and temporal dynamics are encoded in $\mathcal{H}_t$ and $\mathcal{R}_t$, respectively, the output module allows to predict the next snapshot $\hat{\mathcal{S}}_{t+1}$ (see Fig. 4) via MLP: $\hat{\mathcal{S}}_{t+1} = MLP([\mathcal{H}_t, \mathcal{R}_t])$. Because it is impractical to train the entire TTS at once [36], a sliding window strategy [16] – composed of w historical and τ future steps – is used for the training process. In this way, the objective function to solve is given by:

$$\begin{aligned} \underset{\Theta,W,B}{\arg\min} \; & \| N_E T^3\left(\{\mathcal{S}_{t'}\}_{t'=t-w+1}^{t} - \{\mathcal{S}_{t'}\}_{t'=t+1}^{t+\tau} \right) \|_F^2 \\ & + \mu_1 \sum_{t'=t-w+1}^{t} \| \mathcal{H}_{t'} - \mathcal{Z}_{t'} \prod_{m=1}^{M} \times_m U_m \|_F^2 \\ & + \mu_2 \sum_{m=1}^{M} \| U_m U_m^T - I_m \|_F^2 \end{aligned} \tag{7}$$

where $\| \cdot \|_F$ is the Frobenius norm, B are the bias vectors, and μ_1 and μ_2 are tuning parameters for the tensor reconstruction error and orthonormality regularization for U_m, respectively. As a result, the τ future steps are recurrently predicted as the sliding window progresses until the entire TTS is analyzed.

4. Case study

In this section, we demonstrate the proposed methodology for the accurate droplet evolution prediction for unseen and seen material/process settings in the IJP process.

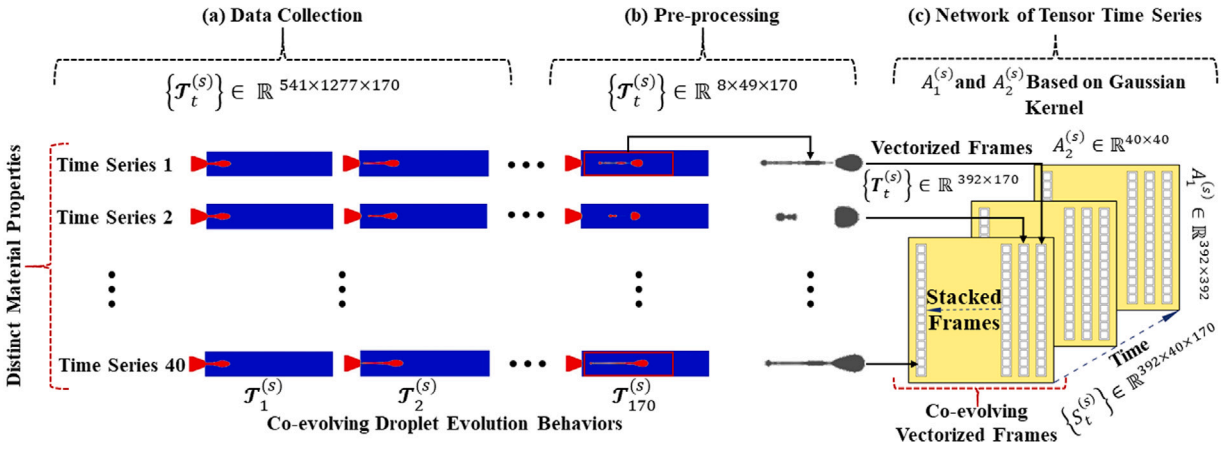


Fig. 5. A Schematic Illustration of the Simulated Data Collection, Pre-processing, and Network of Tensor Time Series Construction.

4.1. Network of tensor time series construction

We collected simulated and experimental video data as specified in Sections 3.2.1 and 3.2.2, respectively. Fig. 5 shows an illustration of the data collection, pre-processing, and network of tensor time series construction for the simulated data. The original simulated $(\mathcal{T}_t^{(s)} \in \mathbb{R}^{541\times1277})$ and experimental $(\mathcal{T}_t^{(e)} \in \mathbb{R}^{480\times640})$ data frames were processed in a similar fashion. During the pre-processing (see Fig. 5(b)), we first converted the original color images to grayscale and then extracted the sub-area corresponding to the shape of the evolving droplet and further downsampled (in terms of resolution) the images in both cases. As a result, we obtained $\mathcal{T}_t^{(s)} \in \mathbb{R}^{8\times49}$ and $\mathcal{T}_t^{(e)} \in \mathbb{R}^{13\times40}$ after the pre-processing. We then vectorized the pixel values (see Fig. 5(c)) of each frame in the different time series (i.e., one time series per set of material/process settings), yielding $\boldsymbol{T}_t^{(s)} \in \mathbb{R}^{392}$ and $\boldsymbol{T}_t^{(e)} \in \mathbb{R}^{520}$, respectively. Subsequently, we formed co-evolving time series for both simulated and experimental droplet evolution cases by stacking the vectorized frames of individual time series corresponding to distinct material/process parameters (see Fig. 5(c)); resulting in $\{\boldsymbol{S}_t^{(s)}\} \in \mathbb{R}^{392\times40\times170}$ and $\{\boldsymbol{S}_t^{(e)}\} \in \mathbb{R}^{520\times20\times173}$. Here, the first dimension corresponds to the vectorized frames, the second dimension is the number of samples (i.e., different material/process settings), and the third dimension indicates the number of time steps. Once the tensor co-evolving time series are established, two networks are formed, namely network of vectorized pixel value locations (e.g., adjacency matrix $A_1^{(s)} \in \mathbb{R}^{392\times392}$) and network of material/process parameters (e.g., adjacency matrix $A_2^{(s)} \in \mathbb{R}^{40\times40}$), as shown in Fig. 5(c). The adjacency matrices are computed as follows:

$$\boldsymbol{A}_{i,j} = \begin{cases} \exp\left(-\frac{\left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|^2}{\sigma}\right) & i \neq j \\ 0 & i = j \end{cases} \tag{8}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are pair of vectors allocating either material/process parameters or pixel value locations. Fig. 5 (c) shows how these networks constrain the within and across material/process parameters pixel values of the different time series. We have $\{\boldsymbol{S}_t^{(s)}\} \in \mathbb{R}^{392\times40\times170}$, $A_1^{(s)} \in \mathbb{R}^{392\times392}$ and $A_2^{(s)} \in \mathbb{R}^{40\times40}$ and $\{\boldsymbol{S}_t^{(e)}\} \in \mathbb{R}^{520\times20\times173}$, $A_1^{(e)} \in \mathbb{R}^{520\times520}$ and $A_2^{(e)} \in \mathbb{R}^{20\times20}$ for the simulated and experimental analysis, respectively.

4.2. Simulated video data prediction results

We first analyze the simulated droplet evolution video data. The presented framework is deployed to perform droplet evolution prediction for unseen and seen material parameters. As for the TGCN and TRNN specifications, we use one layer TGCN and one layer TRNN. The hidden dimension (i.e., d dimension of the node embeddings tensor $\boldsymbol{\mathcal{H}}$ in Eq. (5)) for the hidden features (i.e., TTS cross-linked and spatial relationships) in the TGCN is set to 40. Since TRNN performs faster for lower tensor dimensions, we use 0.3 of the original dimensions for the Tucker decomposition ranks; resulting in [118 12], respectively. Additionally, we specify the window size w to 10, the learning rate of Adam optimizer to 0.01, and the stride size to 1. The reconstruction coefficient μ_1 and orthogonal regularization coefficient μ_2 are $1e^{-7}$ in both cases. Finally, the training process is executed in batches of 60 frames.

4.2.1. Prediction results for unseen material properties

The proposed methodology is able to perform droplet evolution predictions for unseen material/process parameters, as shown in Fig. 6. We randomly split the TTS in 80% training data set, 10% validation data set, and 10% testing data set. Here, the validation data set is used to further tune the model parameters (Θ, W, and B). As an illustration, we perform the prediction at sample 20 with material properties 1600 kg/m^3, 0.0018 kg/m s, and 50 dyn/cm, for density, viscosity, and surface tension, respectively. The prediction results are similar regardless of the sample location in the TTS. Fig. 6 shows the underlying truth (see Fig. 6 (a)) and predicted frames of the unseen sequence for different initialization schemes (see Figs. 6 (b)). For the prediction, we initialize the method with 20 frames: (1) from the droplet evolution sequence with the closest material parameters to the original one (Fig. 6 (b.1)) and (2) from the weighted average droplet evolution sequence, where the weight is controlled by the Oh number (Fig. 6 (b.2)). In particular, we use the Oh number $\left(Oh = \frac{\sqrt{We}}{Re} = \frac{\mu}{\sqrt{\gamma\rho\phi}}\right.$, where μ is viscosity, γ is surface tension, ρ is the density of the fluid, and ϕ is a characteristic length, i.e., the diameter of the nozzle$\left.\right)$ since it relates the viscosity, inertia of the fluid, and surface tension, and is a good indicator of the jettability regimen in the IJP [59].

It can be observed that the unseen droplet evolution is predicted with relatively good accuracy (see Fig. 6(b)). This is achieved by the synergetic action of TGCN and TRNN modules, which make it feasible to capture the cross-linked and spatial–temporal dynamics of the co-evolving droplet evolution time series. Even though the recovered droplets' shapes are alike to the original ones as they evolve, the presence of the static stain (see dark pixels aside the predicted droplet shapes in Fig. 6(b)) disturbs the underlying truth and predicted frames comparison. We presume the stain appears due to two reasons: (1) there are substantially different droplet evolution shapes (see Fig. 5) in the tensor time series that disturb the performance of the TGCN, which is in charge of capturing the relationship among the networks, and (2) the initialization scheme determines how the spatial–temporal relationships among the co-evolving time series are captured at initial stages; in this regard, the weighted average initialization scheme, which involves

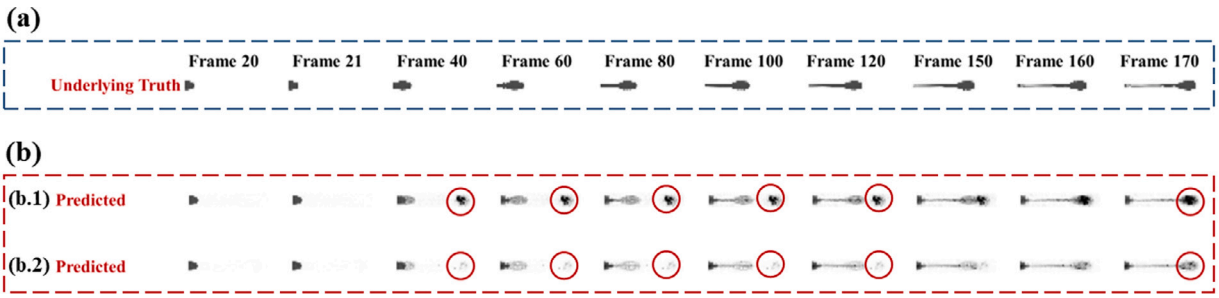


Fig. 6. Unseen Droplet Evolution Sequence Prediction Results: (a) Underlying truth, (b) Predicted frames for different initialization schemes: (b.1) Initialization scheme based on sequence with the closest material parameters to the original one and (b.2) Initialization scheme based on Oh number weighted average.

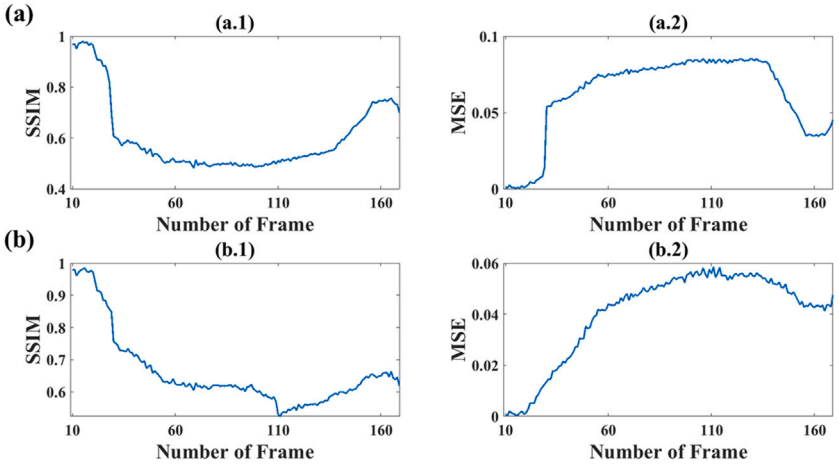


Fig. 7. Unseen Droplet Evolution Sequence Prediction Performance for Different Initialization Schemes: (a) Initialization scheme based on sequence with the closest material parameters to the original one: (a.1) SSIM and (a.2) MSE and (b) Initialization scheme based on Oh number weighted average: (b.1) SSIM and (b.2) MSE.

all the samples in the tensor time series, dissipates better the stain formation as the droplets evolve (see pixel intensity in Fig. 6 (b.2)). The stain disappears at the final frames of the sequence (e.g., frames 150 to 170); resulting in good approximations of the original frames.

Complementary to Fig. 6, we numerically evaluate the prediction performance of unseen material properties via Structural Similarity Index Measurement (SSIM) and Mean Square Error (MSE), as shown in Fig. 7. SSIM performs a pixel-wise comparison between the actual and predicted images. The closer SSIM is to 1, the better the approximation is [60]. MSE is defined as: $MSE = \frac{1}{T}\sum_{t=1}^{T} \| \boldsymbol{\mathcal{T}}_t - \hat{\boldsymbol{\mathcal{T}}}_t \|_F^2$. Figs. 7 (a–b) display the numerical performance for the unseen droplet evolution sequence prediction with different initialization schemes. From the figure, we can observe that the unseen sequence prediction performance decays up to around the 60th frame. Then, the prediction performance stabilizes until the entire unseen sequence is predicted (see Fig. 7).

4.2.2. Prediction results for seen material properties

For the droplet evolution prediction task with observed material properties, we performed one-step and multi-step ahead predictions (Fig. 8). We split the TTS (i.e., co-evolving droplet evolution videos) as follow: the last 10% time steps as test data set and the remaining is distributed for training (80%) and validation (10%) data sets. In Fig. 8, we can observe representative examples of one-step and multi-step ahead predictions of the droplet evolution time series. We randomly pick sample 1 from the TTS, to show the prediction performance of our framework in both cases. In particular, Fig. 8(a) displays the one-step ahead droplet evolution prediction. Here, we can observe accurate prediction by comparing the predicted and underlying truth frames.

We further explore long-term predictions (i.e., several steps ahead) by feeding the network with predicted frames recursively. Fig. 8(b) shows the multi-step ahead prediction of a representative droplet evolution time series. We can observe that the underlying truth and predicted droplet shapes are similar along with all the presented frames (i.e., 1 to 10 steps ahead), which does not occur in our previous study [27]. This is due to the captured synergy from the co-evolving time series via TGCN. Although the droplet shapes are preserved for all the displayed droplet evolution time series, we can also observe that around the 8th-step ahead prediction the droplet is not advancing, and this is more obvious at the 10th-step ahead predictions (see Fig. 8(b)). From this experiment, it is evident that the learned temporal dynamics is depleted as the droplet evolves and no actual observations are present to inform the process that changes occur. Notice that $\boldsymbol{\mathcal{T}}_{t+0}$ is frame 153, which is the last frame of the training data set. $\boldsymbol{\mathcal{T}}_{t+1}$ is the first frame of the testing data (last 10% of the droplet evolution). Accurately anticipating the droplet evolution process gives users/system potential to take correction actions for subsequent droplets, thus preventing the process and products from possible errors and defects.

Similar to the prediction for unseen material properties case, we numerically evaluate the performance of the prediction of seen material properties via SSIM and MSE as shown in Fig. 8(c). In particular, the average SSIM and MSE over all samples are 0.92 (standard deviation 0.04) and 0.0063 (standard deviation 0.0016), respectively, for the one-step ahead predictions. This corroborates the prediction accuracy of the proposed methodology. Furthermore, Fig. 8(c) shows how the SSIM and MSE measures change for multi-step ahead predictions. The SSIM and MSE measures linearly decay and increase, respectively, as the droplet evolves several steps ahead. At the 8th-step ahead prediction the SSIM and MSE are 0.71 and 0.032 (as also indicated in Fig. 8(b)), which can still be considered good for process monitoring and root cause analysis [61]; and beyond this point the prediction performance may not be good.

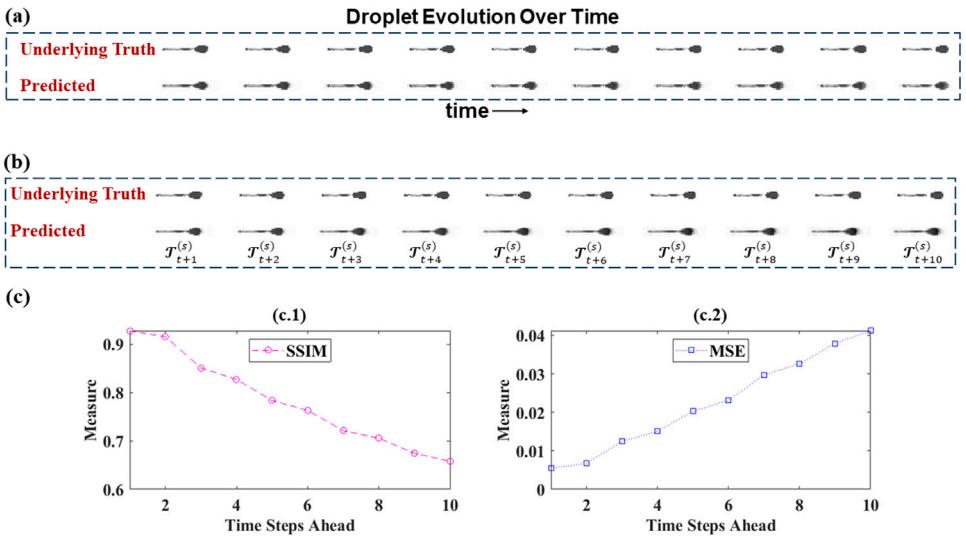


Fig. 8. Seen Droplet Evolution Sequence Prediction Results: (a) One-step ahead prediction sample in the simulated droplet evolution videos, (b) Multi-step ahead prediction sample in the simulated droplet evolution videos, and (c) Prediction performance of multi-step ahead prediction: (c.1) SSIM and (c.2) MSE.

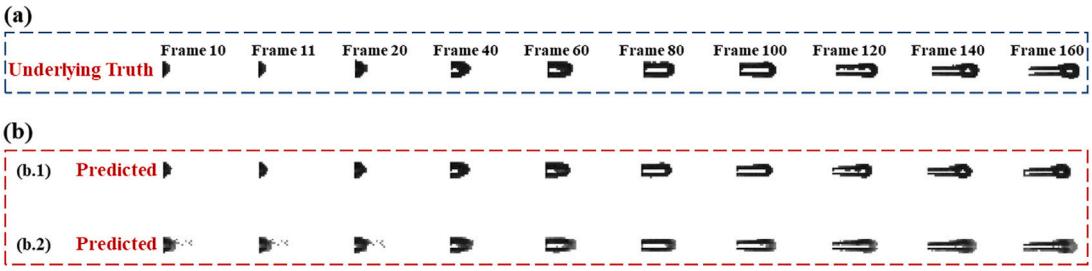


Fig. 9. Experimental Unseen Droplet Evolution Sequence Prediction Results: (a) Underlying truth, (b) Predicted frames for different initialization schemes: (b.1) Initialization scheme based on sequence with the closest material parameters to the original one and (b.2) Initialization scheme based on Oh number weighted average.

4.3. Experimental video data prediction results

Alike to the simulated data prediction, we inputted the experimental network of tensor time series (i.e., $\{\boldsymbol{S}_t^{(e)}\}$, $A_1^{(e)}$ and $A_2^{(e)}$) to the NET^3 framework. Here, we use one layer TGCN and one layer TRNN. The hidden dimensions for the hidden features in the TGCN are set to 40 for the predictions of seen and unseen process settings, respectively. For the Tucker decomposition ranks, we use 0.9 of the original dimensions for unseen and seen scenarios (i.e., [468 18]), respectively. In addition, w, learning rate, stride size, μ_1, μ_2, and batch size are specified as in Section 4.2.

4.3.1. Prediction results for unseen process settings

Similar to Section 4.2.1, we performed droplet evolution predictions for unseen process settings, as shown in Fig. 9. As an illustration, we perform the prediction of sample 10 with process settings −5 in H_2O, 34 μs, 68 μs, 70 V, and −70 V, for back-pressure, dwell time, echo time, dwell voltage, and echo voltage, correspondingly. Fig. 9 shows the underlying truth and predicted frames of the experimental unseen sequence. For the prediction, we initialize the entire unseen droplet evolution sequence with frames: (1) from the droplet evolution sequence with the closest process parameters to the original one (Fig. 9 (b.1)) and (2) from the *Oh* number based weighted average droplet evolution sequence (Fig. 9 (b.2)). Notice that we used the same material (i.e., distilled water) and nozzle diameter (i.e., 100 μm) for all the experiments; thus, the *Oh* number is the same for all the co-evolving time series.

From Fig. 9(b), it can be observed that the experimental unseen droplet evolution frames are predicted with good accuracy. Unlike the simulated unseen scenario prediction in Section 4.2.1, the stain is almost completely dissipated for both initialization schemes; nevertheless, there is a minimal stain at initial stages (see frames 10 to 20 in Fig. 9 (b.1)) of the predicted droplet frames with the Oh number based initialization scheme. This may be imputed to the variable lighting conditions when performing data collection.

Numerical evaluation is also developed via SSIM and MSE, as shown in Fig. 10. For the prediction that is initialized with the closest material parameters to the original one (i.e., unseen sequence), it can be concluded that the prediction performance is steadily decreasing as the number of frames are predicted since only one sample from the entire TTS is considered for the predictions, and this may not represent the overall TTS droplet behaviors (see SSIM and MSE indexes in Figs. 10 (a.1–a.2)). On the other hand, the Oh number based initialization scheme better condenses the information of the overall TTS; resulting in more stable predictions in the long term (see SSIM and MSE indexes in Figs. 10 (b.1–b.2)).

4.3.2. Prediction results for seen process settings

We performed one-step and multi-step ahead predictions (see Fig. 11) in a similar fashion as in Section 4.2.2. Here, we randomly chose sample 5 for the predictions evaluation. In Fig. 11(a), we can observe that the model accurately (i.e., overall SSIM and MSE are 0.96 (standard deviation 0.03) and 0.0062 (standard deviation 0.0041), respectively) predicts one frame ahead of the current observation. In addition, Fig. 11(b) displays examples of underlying truth and predicted frames for the multi-step ahead predictions of a representative droplet evolution time series. Similar to what occurred in the simulated multi-step ahead predictions, the predicted experimental

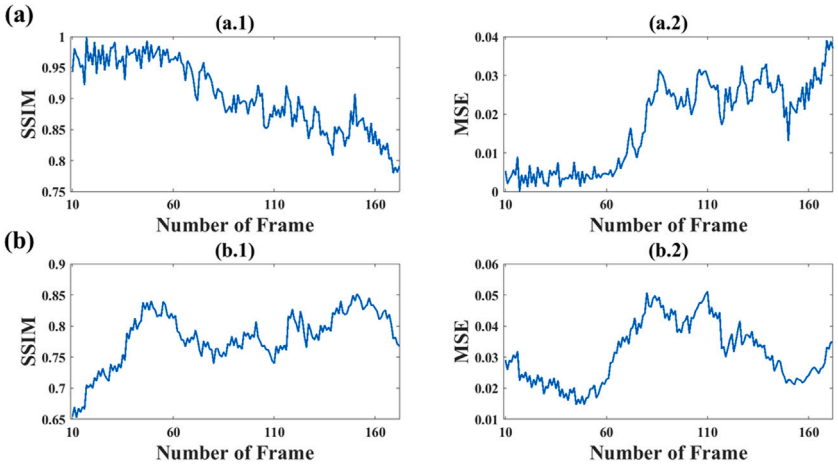


Fig. 10. Experimental Unseen Droplet Evolution Sequence Prediction Performance for Different Initialization Schemes: (a) Initialization scheme based on sequence with the closest material parameters to the original one: (a.1) SSIM and (a.2) MSE and (b) Initialization scheme based on Oh number weighted average: (b.1) SSIM and (b.2) MSE.

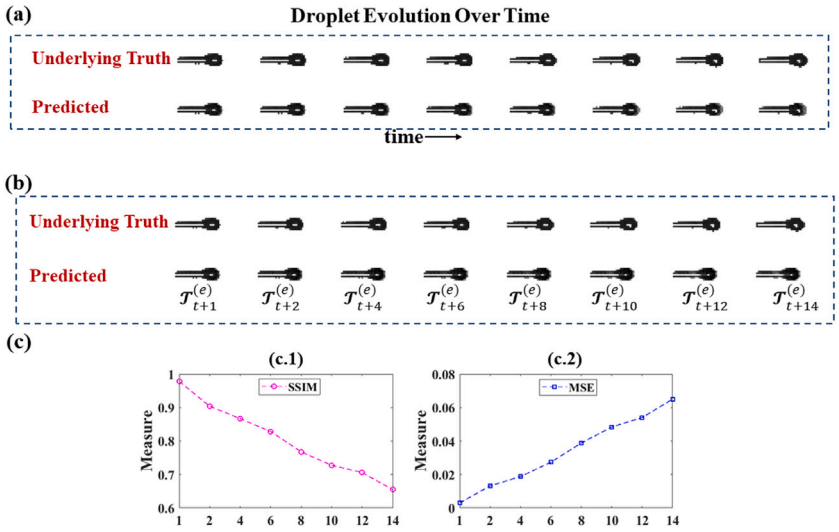


Fig. 11. Experimental Seen Droplet Evolution Sequence Prediction Results: (a) One-step ahead prediction sample in the experimental droplet evolution videos, (b) Multi-step ahead prediction sample in the experimental droplet evolution videos, and (c) Prediction performance of experimental multi-step ahead prediction: (c.1) SSIM and (c.2) MSE.

droplet evolution suffers a delay that starts around the 8th-step ahead and is more evident at the 14th-step ahead predictions (see Fig. 11(b)). Notice that $\mathcal{T}_{t+0}$ is frame 156, which is the last frame of the training data set. $\mathcal{T}_{t+1}$ is the first frame of the testing data (last 10% of the droplet evolution). Fig. 11(c) shows, in terms of SSIM and MSE, how the captured temporal dynamics deteriorates as the predicted droplets evolve. The droplet shapes are conserved during the full evolution process (see Fig. 11(b)). In particular, the 8th-step ahead SSIM and MSE are 0.77 and 0.0389, respectively. The numerical and visual evaluation reveal that the proposed framework is able to perform one-step and multi-step ahead predictions in the IJP droplet evolution experimental video data.

Finally, the proposed framework is able to perform analysis for new scenarios very fast (see Table 3) compared to CFD analysis, where each simulation takes on average 1584.90 s (standard deviation 16.19 s) and only simulated future prediction scenarios are possible. The simulations were performed in a desktop with the following characteristics: Intel(R) Xeon(R) W-2145 CPU @ 3.70 GHz and 64.0 GB of RAM. In summary, the proposed framework can accurately and rapidly perform droplet evolution predictions in the IJP process.

Table 3
Average (Standard Deviation) computational time of network of tensor time series predictions for different scenarios.

Scenario	Time (s)/Predicted frame
Simulated seen prediction	3.49 s (0.11)
Simulated unseen prediction	3.48 s (0.10)
Experimental seen prediction	4.36 s (0.17)
Experimental unseen prediction	2.76 s (0.08)

4.4. Proposed framework applications

Several applications have been identified from our proposed unseen and seen droplet evolution prediction framework. In particular, the unseen droplet evolution prediction scenario learns the relationships between material/process parameters and droplet evolution behavior, which supports printing exploration without the need to do time-consuming physical simulations and experimentations. The prediction also helps ink materials design. For instance, Nallan et al. [62] showed that stable droplet formation and jettability are crucial for the IJP process, and takes a lot of resources (e.g., time and experiments) to

explore the space of ink material properties (e.g., density, viscosity, and surface tension). Our framework opens the possibility of extensive material properties exploration with minimal effort. On the other hand, the seen droplet evolution prediction scenario can accurately forecast the future droplet evolution (e.g., one-step and multi-step ahead predictions), compared with prior work in [27]. With the forecast results, one can perform root cause analysis to identify the process problems by comparing the forecast and actual videos [61].

5. Conclusion and discussion

IJP is able to produce complicated structures with high-quality and high-resolution. Nevertheless, the part quality and process performance can be harmed by the instability of the different droplet morphologies and behaviors during the printing process. These instabilities mainly occur due to different material and process parameters, and studying all the parameter combinations is time-consuming. In this paper, we propose a framework that is capable of making droplet morphologies and behaviors predictions for seen and unseen scenarios. This is achieved by the systematic capture of the cross-linked, spatial and temporal relationships of the co-evolving time series (i.e., droplet evolution behaviors with different material/process parameters). The proposed framework shows promising results at making accurate predictions for simulated and experimental IJP droplet evolution videos. The proposed framework can also be applied to a broad range of material/process parameters as long as the forming mechanisms (i.e., NS principles and/or IJP setup) are preserved.

There are several research directions that can be pursued in the future. One direction is to further increase the prediction accuracy by incorporating generative adversarial networks to generate plausible droplet evolution frames to improve the training process. Another direction is to incorporate a graph/network to study ambient parameters' (e.g., temperature and humidity) influence on the droplet deposition.

CRediT authorship contribution statement

Luis Javier Segura: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Zebin Li:** Writing – review & editing, Writing – original draft. **Chi Zhou:** Writing – review & editing, Resources, Project administration, Funding acquisition, Conceptualization. **Hongyue Sun:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is partially supported by the National Science Foundation, United States Grant No. CMMI-1846863, FM-2134409, Sustainable Manufacturing and Advanced Robotics Technologies, United States, Community of Excellence (SMART CoE) at the State University of New York at Buffalo, United States, Jon Rieger Seed Grant at the University of Louisville, and KY NSF EPSCoR Emerging Research Ideas (ERI) Program, United States.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.addma.2023.103461.

References

[1] D. Lin, S. Jin, F. Zhang, C. Wang, Y. Wang, C. Zhou, G.J. Cheng, 3D stereolithography printing of graphene oxide reinforced complex architectures, Nanotechnology 26 (43) (2015) 434003.
[2] F. Zhang, 3D Freeze Nano Printing for Multiscale, Multifunctional Porous Materials, State University of New York at Buffalo, 2018.
[3] T.D. Ngo, A. Kashani, G. Imbalzano, K.T. Nguyen, D. Hui, Additive manufacturing (3D printing): A review of materials, methods, applications and challenges, Composites B 143 (2018) 172–196.
[4] T. Chu, H. Wang, Y. Qiu, H. Luo, B. He, B. Wu, B. Gao, 3D printed smart silk wearable sensors, Analyst 146 (5) (2021) 1552–1558.
[5] C. Milroy, A. Manthiram, Printed microelectrodes for scalable, high-areal-capacity lithium–sulfur batteries, Chem. Commun. 52 (23) (2016) 4282–4285.
[6] P. Ahangar, M.E. Cooke, M.H. Weber, D.H. Rosenzweig, Current biomedical applications of 3D printing and additive manufacturing, Appl. Sci. 9 (8) (2019) 1713.
[7] X. Fang, J. Shi, X. Zhang, X. Ren, B. Lu, W. Deng, J. Jie, X. Zhang, Patterning liquid crystalline organic semiconductors via inkjet printing for high-performance transistor arrays and circuits, Adv. Funct. Mater. 31 (21) (2021) 2100237.
[8] D.S. Kolchanov, I. Mitrofanov, A. Kim, Y. Koshtyal, A. Rumyantsev, E. Sergeeva, A. Vinogradov, A. Popovich, M.Y. Maximov, Inkjet printing of Li-rich cathode material for thin-film lithium-ion microbatteries, Energy Technol. 8 (3) (2020) 1901086.
[9] F. Zheng, J. Jang, C. Tse, J. Brugger, 3D printed micro-scaffolds loaded by inkjet printing with μg-precise amount of drug, in: 2020 IEEE 15th International Conference on Nano/Micro Engineered and Molecular System, NEMS, IEEE, 2020, pp. 426–429.
[10] S.D. Hoath, Fundamentals of Inkjet Printing: The Science of Inkjet and Droplets, John Wiley & Sons, 2016.
[11] M.-H. Tsai, W.-S. Hwang, Effects of pulse voltage on the droplet formation of alcohol and ethylene glycol in a piezoelectric inkjet printing process with bipolar pulse, Mater. Trans. 49 (2) (2008) 331–338.
[12] T. Wang, C. Zhou, W. Xu, Online droplet monitoring in inkjet 3D printing using catadioptric stereo system, IISE Trans. 51 (2) (2019) 153–167.
[13] D. Zhao, H. Zhou, Y. Wang, J. Yin, Y. Huang, Drop-on-demand (DOD) inkjet dynamics of printing viscoelastic conductive ink, Addit. Manuf. 48 (2021) 102451.
[14] A. van der Bos, M.-J. van der Meulen, T. Driessen, M. van den Berg, H. Reinten, H. Wijshoff, M. Versluis, D. Lohse, Velocity profile inside piezoacoustic inkjet droplets in flight: comparison between experiment and numerical simulation, Phys. Rev. A 1 (1) (2014) 014004.
[15] B. He, S. Yang, Z. Qin, B. Wen, C. Zhang, The roles of wettability and surface tension in droplet formation during inkjet printing, Sci. Rep. 7 (1) (2017) 1–7.
[16] L.J. Segura, T. Wang, C. Zhou, H. Sun, Online droplet anomaly detection from streaming videos in inkjet printing, Addit. Manuf. 38 (2021) 101835.
[17] L.J. Segura, C.N. Muñoz, C. Zhou, H. Sun, Sketch-based tensor decomposition for non-parametric monitoring of electrospinning processes, in: International Manufacturing Science and Engineering Conference, vol. 84263, American Society of Mechanical Engineers, 2020, V002T09A002.
[18] T. Jiao, Q. Lian, T. Zhao, H. Wang, Influence of ink properties and voltage parameters on piezoelectric inkjet droplet formation, Appl. Phys. A 127 (1) (2021) 1–9.
[19] T.-M. Liou, C.-Y. Chan, K.-C. Shih, Effects of actuating waveform, ink property, and nozzle size on piezoelectrically driven inkjet droplets, Microfluid. Nanofluid. 8 (5) (2010) 575–586.
[20] Y. Guo, H.S. Patanwala, B. Bognet, A.W. Ma, Inkjet and inkjet-based 3D printing: connecting fluid properties and printing performance, Rapid Prototyping Journal (2017).
[21] A. Lee, K. Sudau, K.H. Ahn, S.J. Lee, N. Willenbacher, Optimization of experimental parameters to suppress nozzle clogging in inkjet printing, Ind. Eng. Chem. Res. 51 (40) (2012) 13195–13204.
[22] J. Tai, H.Y. Gan, Y.N. Liang, B.K. Lok, Control of droplet formation in inkjet printing using Ohnesorge number category: materials and processes, in: 2008 10th Electronics Packaging Technology Conference, IEEE, 2008, pp. 761–766.
[23] E. Kim, J. Baek, Numerical study on the effects of non-dimensional parameters on drop-on-demand droplet formation dynamics and printability range in the up-scaled model, Phys. Fluids 24 (8) (2012) 082103.
[24] H.-C. Wu, H.-J. Lin, W.-S. Hwang, A numerical study of the effect of operating parameters on drop formation in a squeeze mode inkjet device, Modelling Simul. Mater. Sci. Eng. 13 (1) (2004) 17.
[25] A. Fakhari, M.H. Rahimian, Simulation of falling droplet by the lattice Boltzmann method, Commun. Nonlinear Sci. Numer. Simul. 14 (7) (2009) 3046–3055.

[26] X.Q. Xing, D.L. Butler, S.H. Ng, Z. Wang, S. Danyluk, C. Yang, Simulation of droplet formation and coalescence using lattice Boltzmann-based single-phase model, J. Colloid Interface Sci. 311 (2) (2007) 609–618.
[27] J. Huang, L.J. Segura, T. Wang, G. Zhao, H. Sun, C. Zhou, Unsupervised learning for the droplet evolution prediction and process dynamics understanding in inkjet printing, Addit. Manuf. 35 (2020) 101197.
[28] H. Yao, X. Tang, H. Wei, G. Zheng, Y. Yu, Z. Li, Modeling spatial-temporal dynamics for traffic prediction, 2018, arXiv preprint arXiv:1803.01254.
[29] F. Liu, Y. Deng, A fast algorithm for network forecasting time series, IEEE Access 7 (2019) 102554–102560.
[30] R. Han, Y. Wang, Y. Zhang, G. Chen, A novel spatial-temporal prediction method for unsteady wake flows based on hybrid deep neural network, Phys. Fluids 31 (12) (2019) 127101.
[31] X. Jin, P. Cheng, W.-L. Chen, H. Li, Prediction model of velocity field around circular cylinder over various Reynolds numbers by fusion convolutional neural networks based on pressure on the cylinder, Phys. Fluids 30 (4) (2018) 047105.
[32] V. Sekar, Q. Jiang, C. Shu, B.C. Khoo, Fast flow field prediction over airfoils using deep learning approach, Phys. Fluids 31 (5) (2019) 057103.
[33] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, et al., Deep learning and process understanding for data-driven Earth system science, Nature 566 (7743) (2019) 195–204.
[34] Y. Chang, Z. Tu, W. Xie, B. Luo, S. Zhang, H. Sui, J. Yuan, Video anomaly detection with spatio-temporal dissociation, Pattern Recognit. 122 (2022) 108213.
[35] S. Wang, J. Cao, P. Yu, Deep learning for spatio-temporal data mining: A survey, IEEE Trans. Knowl. Data Eng. (2020).
[36] B. Jing, H. Tong, Y. Zhu, Network of tensor time series, in: Proceedings of the Web Conference 2021, 2021, pp. 2425–2437.
[37] D. Wu, C. Xu, Predictive modeling of droplet formation processes in inkjet-based bioprinting, J. Manuf. Sci. Eng. 140 (10) (2018).
[38] Y. Pan, M. Hu, A data-driven modeling approach for digital material additive manufacturing process planning, in: 2016 International Symposium on Flexible Automation, ISFA, IEEE, 2016, pp. 223–228.
[39] S. Stoyanov, C. Bailey, Machine learning for additive manufacturing of electronics, in: 2017 40th International Spring Seminar on Electronics Technology, ISSE, IEEE, 2017, pp. 1–6.
[40] S. Andalib, K. Taira, H.P. Kavehpour, Data-driven time-dependent state estimation for interfacial fluid mechanics in evaporating droplets, Sci. Rep. 11 (1) (2021) 1–11.
[41] C. Tian, T. Li, J. Bustillos, S. Bhattacharya, T. Turnham, J. Yeo, A. Moridi, Data-driven approaches toward smarter additive manufacturing, Adv. Intell. Syst. (2021) 2100014.
[42] G. Tourloukis, S. Stoyanov, T. Tilford, C. Bailey, Predictive modelling for 3D inkjet printing processes, in: 2016 39th International Spring Seminar on Electronics Technology, ISSE, IEEE, 2016, pp. 257–262.
[43] U. Inyang-Udoh, S. Mishra, A learning-based approach to modeling and control of inkjet 3D printing, in: 2020 American Control Conference, ACC, IEEE, 2020, pp. 460–466.
[44] J. Shi, J. Song, B. Song, W.F. Lu, Multi-objective optimization design through machine learning for drop-on-demand bioprinting, Engineering 5 (3) (2019) 586–593.
[45] M. Ogunsanya, J. Isichei, S.K. Parupelli, S. Desai, Y. Cai, In-situ droplet monitoring of inkjet 3D printing process using image analysis and machine learning models, Procedia Manuf. 53 (2021) 427–434.
[46] A.K. Ball, R. Das, S.S. Roy, D.R. Kisku, N.C. Murmu, Modeling of EHD inkjet printing performance using soft computing-based approaches, Soft Comput. 24 (1) (2020) 571–589.
[47] H. Zhang, S.K. Moon, Reviews on machine learning approaches for process optimization in noncontact direct ink writing, ACS Appl. Mater. Interfaces (2021).
[48] N.K. Ahmed, A.F. Atiya, N.E. Gayar, H. El-Shishiny, An empirical comparison of machine learning models for time series forecasting, Econometric Rev. 29 (5–6) (2010) 594–621.
[49] L. Kirichenko, T. Radivilova, V. Bulakh, Machine learning in classification time series with fractal properties, Data 4 (1) (2019) 5.
[50] B.M. Pavlyshenko, Linear, machine learning and probabilistic approaches for time series analysis, in: 2016 IEEE First International Conference on Data Stream Mining & Processing, DSMP, IEEE, 2016, pp. 377–381.
[51] Z. Shen, Y. Zhang, J. Lu, J. Xu, G. Xiao, A novel time series forecasting model with deep learning, Neurocomputing 396 (2020) 302–313.
[52] W. Bao, J. Yue, Y. Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, PLoS One 12 (7) (2017) e0180944.
[53] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, H. Zhang, Deep learning with long short-term memory for time series prediction, IEEE Commun. Mag. 57 (6) (2019) 114–119.
[54] J.C.B. Gamboa, Deep learning for time-series analysis, 2017, arXiv preprint arXiv:1701.01887.
[55] B. Lim, S. Zohren, Time-series forecasting with deep learning: a survey, Phil. Trans. R. Soc. A 379 (2194) (2021) 20200209.
[56] G. Bontempi, S.B. Taieb, Y.-A. Le Borgne, Machine learning strategies for time series forecasting, in: European Business Intelligence Summer School, Springer, 2012, pp. 62–77.
[57] G.E. Box, G.M. Jenkins, G.C. Reinsel, G.M. Ljung, Time Series Analysis: Forecasting and Control, John Wiley & Sons, 2015.
[58] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.
[59] C. Xu, Z. Zhang, Y. Huang, H. Xu, Phase diagram of pinch-off behaviors during drop-on-demand inkjetting of alginate solutions, J. Manuf. Sci. Eng. 141 (9) (2019) 091013.
[60] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[61] V.L. Miguéis, J.L. Borges, et al., Automatic root cause analysis in manufacturing: an overview & conceptualization, J. Intell. Manuf. (2022) 1–18.
[62] H.C. Nallan, J.A. Sadie, R. Kitsomboonloha, S.K. Volkman, V. Subramanian, Systematic design of jettable nanoparticle-based inkjet inks: Rheology, acoustics, and jettability, Langmuir 30 (44) (2014) 13470–13477.